

# Riverbed Cascade Profiler REST API. v1.2

Copyright © Riverbed Technology Inc. 2024

Created Jan 16, 2024 at 02:01 PM

## Overview

The documentation pages in this section describe the RESTful APIs included with Cascade Profiler and Cascade Express products. It is assumed that the reader has practical knowledge of RESTful APIs, so the documentation does not go into detail about what REST is and how to use it. Instead the documentation focuses on what data can be accessed and how to access it.

The primary focus of the current version of the API is on providing access to reporting data. The following information can be accessed via the API:

- Reporting data (traffic data, service data, event list data, and user data)
- Information about network devices
- Control over SDN virtual network settings (SDN virtual network settings are a licensed feature)
- Information about system users

Details about REST resources can be found in the **Resources** section. This overview continues with how to run reports and retrieve data from them.

## Running a report

The steps for running a report are described below. An easy way to learn how to run reports and get data is to use the **Demo Reports** section of this documentation. In that section a number of example reports are listed. If you click on any of the examples, the report will run along with a listing on the right side of the screen for each step in the process. It displays the REST resource, HTTP method, headers and body for both the request and response.

Follow these steps to run a report:

### 1. Create a new report by posting report criteria to the server.

A criteria structure in either JSON or XML is submitted to the server using the HTTP POST method. The resource to which the criteria structure is posted is called **/profiler/1.0/reporting/reports**. The details are described in the **Resources** section of this documentation.

A key part of the report criteria is the ID of the template that should be used to run the report. A special system template ID 184 that provides a high degree of flexibility is used in demo reports in this documentation. Additionally, any template that is saved via the user interface of the product can be used to run a report. In order to save a template, the ID of that template must be passed in the report criteria structure instead of 184. A template can be configured via the user interface, saved via the product and then used in the REST API to generate reports and retrieve them in a rendered form or in raw data form. Once a template is saved, its ID can be obtained via the **/api/profiler/1.0/reporting/templates** REST resource.

### 2. Poll report status until the report completes.

It may take a while for a report to complete. While the report is running, the client must poll for report status to know when the report completes. When the call to create a new report succeeds, it returns the URL for the newly created report. That URL may look similar to */profiler/1.0/reporting/reports/1000*, which is the ID of the new report.

The general way to describe this in REST documentation is **/profiler/1.0/reporting/reports/{id}** so this documentation uses that syntax throughout. Note that the client does not need to know that **{id}** is really an ID of a report. Instead the client should treat a given report, for example */profiler/1.0/reporting/reports/1000*, as a REST resource without parsing the parts of the URL.

The status of a report can be obtained by executing a GET method on the report URL. The client should wait while the report is running and until the status turns to state.

### 3. Retrieve the report data.

Once the report completes, the client can retrieve its data or the rendered version of the report in a number of formats.

The following resources can be used to retrieve a rendered version of the report:

- */profiler/1.0/reporting/reports/{id}/view.pdf*
- */profiler/1.0/reporting/reports/{id}/view.csv*

These are for PDF and CSV versions respectively.

If the client is a script that needs access to raw data, the **/profiler/1.0/reporting/reports/{id}/queries** resource can be used with the GET method to obtain the list of elements (queries) first. The data shown in a typical report on the user interface may come from more than one query, which is why the report structure needs to be determined first. However, the system template 184 described above will have only one query and is easy to use for simple scripts.

Each query resource provides metadata about the query, such as the list of columns with descriptions of what the columns are.

Once the query is chosen, the **/profiler/1.0/reporting/reports/{id}/queries/{id}** resource can be used to get the report data.

The simple overview provided above cannot substitute for full documentation and it is not intended to do so. Please refer to **Demo Reports** section to see how reports are run. Look at the **Coding Examples** under **General Information** and explore **Resources** section of this documentation for more information.

---

## Authentication

All REST requests must be authenticated. The **Authentication** section of the Common 1.0 API describes which authentication methods are presently supported. There are also examples that show how to use each of the different authentication methods.

---

## /api/profiler/1.1 to /api/profiler/1.2 Changelog

### Added the following resources:

- /api/profiler/1.2/applications
- /api/profiler/1.2/applications/layer\_4
- /api/profiler/1.2/applications/layer\_7
- /api/profiler/1.2/dscps
- /api/profiler/1.2/host\_group\_types
- /api/profiler/1.2/interfaces
- /api/profiler/1.2/port\_groups
- /api/profiler/1.2/port\_names
- /api/profiler/1.2/protocols
- /api/profiler/1.2/sharks
- /api/profiler/1.2/steelheads
- /api/profiler/1.2/devices/restsync

For more information please read the Resources documentation.

### Added the following methods to existing resources:

- Multiple methods to configure and export Dashboards added under: /api/profiler/1.2/reporting/templates
- DELETE /api/profiler/1.2/devices/{IP}
- POST /api/profiler/1.2/users/radius/test\_user

For more information please read the Resources documentation.

---

## Running a report: Sample PHP script

Run a report to get bytes and packets for the top 20 hosts using the application WEB. Use BASIC Authentication.

```
<?php

define('HOST', '127.0.0.1'); // IP address of Profiler
define('BASIC_AUTH', 'admin:admin');

// Timeframe
$end = time() - 3*60;
$start = $end - 5*60;

// Lib functions

// HTTP POST
function do_POST($url, $string, &$info) {
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_HTTPAUTH, CURLAUTH_BASIC );
```

```

curl_setopt($curl, CURLOPT_USERPWD, BASIC_AUTH);
curl_setopt($curl, CURLOPT_SSLVERSION,3);
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 2);
curl_setopt($curl, CURLOPT_HEADER, true);
curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
curl_setopt($curl, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
curl_setopt($curl, CURLOPT_POST, 1);
curl_setopt($curl, CURLOPT_POSTFIELDS, $string);
$output = curl_exec($curl);
$info = curl_getinfo($curl);
curl_close($curl);

$headers = substr($output, 0, $info['header_size']);
$headers = explode("\n", $headers);
$info['headers'] = $headers;
$body = substr($output, $info['header_size']);
return $body;
}

// HTTP GET
function do_GET($url, &$info) {
    $curl = curl_init();
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_HTTPAUTH, CURLAUTH_BASIC );
    curl_setopt($curl, CURLOPT_USERPWD, BASIC_AUTH);
    curl_setopt($curl, CURLOPT_SSLVERSION,3);
    curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, FALSE);
    curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 2);
    curl_setopt($curl, CURLOPT_HEADER, true);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($curl, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
    curl_setopt($curl, CURLOPT_HTTPGET, true);
    $output = curl_exec($curl);
    $info = curl_getinfo($curl);
    curl_close($curl);

    $headers = substr($output, 0, $info['header_size']);
    $headers = explode("\n", $headers);
    $info['headers'] = $headers;
    $body = substr($output, $info['header_size']);
    return $body;
}

// Finds and returns HTTP header
function get_header($headers, $header) {
    foreach($headers as $h) {
        if (strpos($h, $header . ':') !== false)
            return trim(substr($h, 10));
    }
    echo "Unable to find {$header} header!\n";
    exit;
}

// Locates a column by id and returns the name
function find_column_name_by_id($columns, $id) {
    foreach ($columns as $c) {
        if ($c['id'] == $id)
            return $c['name'];
    }
    return 'Unknown';
}

// CSV helper
function echo_csv($headers, $rows) {
    echo implode(',', $headers) . "\n";
    foreach ($rows as $row)
        echo implode(',', $row) . "\n";
}

// End lib functions

$struct =
    array('template_id' => 184,
        'criteria' => Array('time_frame' => array('start' => $start,
            'end' => $end),
            'traffic_expression' => 'app WEB',
            'query' => array('realm' => 'traffic_summary',
                'group_by' => 'hos',
                'sort_column' => 33,
                'columns' => array(6, 33, 34))););

```

```

$json = json_encode($struct);
$columns = $struct['criteria']['query']['columns'];

// Post to run the report
$url = 'https://' . HOST . '/api/profiler/1.0/reporting/reports.json';
echo "Run report:\nPOST {$url}\n{$json}\n\n";
$info = array();
do_POST($url, $json, $info);
if ($info['http_code'] != 201) {
    echo "Unable to run report!\n";
    exit(1);
}
$location = get_header($info['headers'], 'Location');
echo "Generated: {$location}\n\n";
$status_url = 'https://' . HOST . " . $location . '.json';

// Wait for it to complete
echo "Please wait\n";
while (true) {
    $info = array();
    $output = do_GET($status_url, $info);
    $s = json_decode($output, 1);
    print " Percent completed {$s['percent']}, seconds remaining {$s['remaining_seconds']}...\n";
    if ($s['status'] == 'completed') {
        echo "Completed\n\n";
        break;
    }
    sleep(1);
}

// Get all queries (In this exampe it is only one)
$queries_url = 'https://' . HOST . " . $location . '/queries.json';
$output = do_GET($queries_url, $info);
$queries = json_decode($output, 1);

// Print the data from all queries
foreach ($queries as $q) {
    $query_id = $q['id'];
    $data_url = 'https://' . HOST . " . $location . '/queries/' . $query_id . '.json?offset=0&limit=20&columns=' . join(',', $columns);
    $info = array();
    $output = do_GET($data_url, $info);
    $data = json_decode($output, 1);

    $h = array();
    foreach ($columns as $c)
        $h[] = find_column_name_by_id($q['columns'], $c);

    echo_csv($h, $data['data']);
    echo "\n";
}

?>

```

## Running a report: Sample Python script

Run a report to get bytes and packets for the top 20 hosts using the application WEB. Use BASIC Authentication.

```

from urlparse import urlparse
import base64
import logging
import httplib
import json
import time
import sys

HOST = '127.0.0.1'
BASIC_AUTH = 'admin:admin'

end = int(time.time() - 3*60)
start = int(end - 5*60);

# Lib functions

def do_POST(url, string):
    """HTTP POST"""

    conn = httplib.HTTPSConnection(HOST, 443)

    headers = {"Authorization" : "Basic %s" % base64.b64encode(BASIC_AUTH),

```

```

        "Content-Length" : str(len(string)),
        "Content-Type" : "application/json"}

conn.request('POST', url, body=string, headers=headers)

res = conn.getresponse()

info = {"status" : res.status,
        "headers" : res.getheaders()}

data = res.read()
conn.close()
return data, info

def do_GET(url):
    """HTTP GET"""

    conn = httplib.HTTPSConnection(HOST, 443)

    headers = {"Authorization" : "Basic %s" % base64.b64encode(BASIC_AUTH),
               "Content-Length" : 0,
               "Content-Type" : "application/json"}

    conn.request('GET', url, body="", headers=headers)

    res = conn.getresponse()

    info = {"status" : res.status,
            "headers" : res.getheaders()}

    data = res.read()
    conn.close()
    return data, info

def get_header(headers, header):
    """Finds and returns HTTP header"""
    for i in headers:
        if (i[0] == header):
            return i[1]
    return ""

def find_column_name_by_id(columns, cid):
    """Locates a column by id and returns the name"""
    for c in columns:
        if (c['id'] == cid):
            return c['name']
    return 'Unknown'

def echo_csv(headers, rows):
    """CSV helper"""
    print ', '.join(headers)
    for row in rows:
        print ', '.join(row)

# End lib functions

struct = {
    "template_id" : 184,
    "criteria" : {
        "time_frame" : {
            "start" : start,
            "end" : end
        },
        "traffic_expression" : "app WEB",
        "query" : {
            "realm" : "traffic_summary",
            "group_by" : "hos",
            "sort_column": 33,
            "columns" : [6, 33, 34]
        }
    }
}

to_post = json.dumps(struct)
columns = struct["criteria"]["query"]["columns"]

# Post to run the report
url = "https://%s/api/profiler/1.0/reporting/reports.json" % HOST
print "Run report:"
print "POST %s" % url
print "%s" % to_post

```

```

output, info = do_POST(url, to_post)
if (info['status'] is not 201):
    print "Unable to run report"
    sys.exit(1)

location = get_header(info['headers'], 'location')
print ""
print "Generated: %s" % location
print ""

status_url = "https://%s%s.json" % (HOST, location)

# Wait for it to complete
print "Please wait"
while (True):
    output, info = do_GET(status_url)
    s = json.loads(output)
    print "Percent completed %s, seconds remaining %s..." % (s["percent"], s["remaining_seconds"])
    if (s["status"] == "completed"):
        print "Completed"
        break
    time.sleep(1)

# Get all queries (In this example it is only one)
queries_url = "https://%s%s/queries.json" % (HOST, location)

output, info = do_GET(queries_url)
queries = json.loads(output)

# Print the data from all queries
for q in queries:
    query_id = q['id'];
    columns_str = ','.join([repr(i) for i in columns])
    data_url = "https://%s%s/queries/%s.json?offset=0&limit=20&columns=%s" % (HOST, location, query_id, columns_str)
    output, info = do_GET(data_url)
    data = json.loads(output)

    h = []
    for c in columns:
        h.append(find_column_name_by_id(q["columns"], c))

    print ""
    echo_csv(h, data["data"])

```

## Running a report: Sample Perl script

Run a report to get bytes and packets for the top 20 hosts using the application WEB. Use BASIC Authentication.

```

#!/usr/bin/perl
use strict;
use warnings;

use LWP::UserAgent;
use HTTP::Request;
use List::MoreUtils qw(firstidx);
use JSON qw( encode_json decode_json );

use constant HOST    => '127.0.0.1';
use constant LOGIN   => 'admin';
use constant PASSWORD => 'admin';

our $ua = LWP::UserAgent->new;
$ua->agent("ProfilerScript/0.1");

our $API_BASE = "https://127.0.0.1";

sub _request($)
{
    my $req = shift;

    $req->header('Accept' => 'application/json');
    $req->authorization_basic(LOGIN, PASSWORD);

    my $res = $ua->request($req);

    return {
        code    => $res->code,
        status  => $res->status_line,
        headers => $res->headers()
    };
}

```

```

headers = $res->headers;
data => decode_json($res->content)
};
}

sub GET($)
{
my $req = HTTP::Request->new(GET => $API_BASE . shift);
return _request($req);
}

sub POST($$)
{
my $req = HTTP::Request->new(POST => $API_BASE . shift);
$req->content_type('application/json');
$req->content(encode_json(shift));

return _request($req);
}

my $end = time();
my $start = $end - 5 * 60;

my $struct = {
template_id => 184,
criteria => {
time_frame => {
start => $start,
end => $end
},
traffic_expression => "app WEB",
query => {
realm => "traffic_summary",
group_by => "hos",
sort_column => 33,
columns => [6, 33, 34]
}
}
};

print "Running report... ";

my $response = POST('/api/profiler/1.0/reporting/reports', $struct);
die "Unable to run report. $response->{data}->{error_text}" unless $response->{code} == 201;

my $loc = $response->{headers}->header('Location');

while (1)
{
$response = GET($loc);
printf "\rRunning report, %3d%% done, %d seconds remaining... ",
$response->{data}->{percent},
$response->{data}->{remaining_seconds};

last if $response->{data}->{status} eq 'completed';
sleep(1);
};

print "\nLoading data...\n";

$response = GET($loc . '/queries');
die "Unable to load queries. $response->{data}->{error_text}" unless $response->{code} == 200;

foreach my $query (@{$response->{data}})
{
my @columns = @{$struct->{criteria}->{query}->{columns}};
my $columns = join '|', @columns;

my $data_response = GET("$loc/queries/$query->{id}?offset=0&limit=20&columns=$columns");
die "Unable to load data. $response->{data}->{error_text}" unless $response->{code} == 200;

my @indices = map { my $id = $_; firstidx { $_->{id} == $id } @{$query->{columns}} } @columns;

print join ",", map { qq~"$query->{columns}->[$_]->{name}"~; } @indices;
print "\n";

foreach my $row (@{$data_response->{data}->{data}}) {
print join ",", @$row;
print "\n";
}
}
}

```



## Running a report: Sample .Net/C# code

Run a report to get bytes and packets for the top 20 hosts using the application WEB. Use BASIC Authentication.

### Program.cs:

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Runtime.Serialization.Json;
using System.Text;
using System.IO;
using System.Net.Security;
using System.Security.Cryptography.X509Certificates;
using System.Linq;
using System.Threading;
using System.Web.Script.Serialization;

namespace CascadeRestClient
{
    public class ReportUpdate
    {
        public string status { get; set; }
        public string user_id { get; set; }
        public string name { get; set; }
        public string percent { get; set; }
        public string id { get; set; }
        public string remaining_seconds { get; set; }
        public string run_time { get; set; }
        public string saved { get; set; }
        public string template_id { get; set; }
        public string size { get; set; }
    }

    public class Column
    {
        public string strid { get; set; }
        public string metric { get; set; }
        public string rate { get; set; }
        public string statistic { get; set; }
        public int id { get; set; }
        public string unit { get; set; }
        public string category { get; set; }
        public string severity { get; set; }
        public string area { get; set; }
        public bool @internal { get; set; }
        public string role { get; set; }
        public string cli_srv { get; set; }
        public string type { get; set; }
        public bool available { get; set; }
        public string direction { get; set; }
        public string comparison { get; set; }
        public bool sortable { get; set; }
        public string name { get; set; }
        public string comparison_parameter { get; set; }
        public bool has_others { get; set; }
        public bool context { get; set; }
        public string name_type { get; set; }
    }

    public class QueryResult
    {
        public string direction { get; set; }
        public string actual_log { get; set; }
        public int actual_t0 { get; set; }
        public bool sort_desc { get; set; }
        public string area { get; set; }
        public string metric { get; set; }
        public int sort_col { get; set; }
        public string parent_id { get; set; }
        public string rate { get; set; }
        public string group_by { get; set; }
        public string role { get; set; }
        public List<Column> columns { get; set; }
        public string statistic { get; set; }
        public string type { get; set; }
        public string id { get; set; }
        public string unit { get; set; }
        public int actual_t1 { get; set; }
    }
}
```

```

public class QueryData
{
    public List<List<string>> data { get; set; }
    public int data_size { get; set; }
    public List<string> totals { get; set; }
}

class Program
{
    static string BASIC_AUTH = "admin:admin";

    // callback used to validate the self-gen certificate in an SSL conversation
    private static bool ValidateRemoteCertificate(object sender, X509Certificate cert, X509Chain chain, SslPolicyErrors policyErrors)
    {
        return true;
        /*
        X509Certificate2 cert2 = new X509Certificate2(cert);
        if (cert2.GetNameInfo(X509NameType.SimpleName, true) == "www.riverbed.com")
            return true;

        return false;
        */
    }

    static void Main(string[] args)
    {
        if (args.Length == 0 || string.IsNullOrEmpty(args[0]))
        {
            Console.WriteLine("Usage: CascadeRestClient hostname");
            return;
        }
        try
        {
            //Code to allow run with self-signed certificates
            // validate cert by calling a function
            ServicePointManager.ServerCertificateValidationCallback += new RemoteCertificateValidationCallback(ValidateRemoteCertificate);

            //Starting to run rest
            string rootUrl = "https://" + args[0];
            string requestUrl = rootUrl + "/api/profiler/1.0/reporting/reports.json";
            string location;

            int start = (int)((DateTime.Now - new DateTime(1970, 1, 1).ToLocalTime()).TotalSeconds) - 8*60; //8 minutes before in unix time
            int end = start + 5*60; //3 minutes before in unix time

            var jsondata = new
            {
                template_id = 184,
                criteria = new
                {
                    time_frame = new
                    {
                        start = start,
                        end = end
                    },
                    traffic_expression = "app WEB",
                    query = new
                    {
                        realm = "traffic_summary",
                        group_by = "hos",
                        sort_column = 33,
                        columns = new List<int> { 6, 33, 34 }
                    }
                }
            };

            //Serialize anonymous type to json
            JavaScriptSerializer serializer = new JavaScriptSerializer();
            string postData = serializer.Serialize(jsondata);

            Console.WriteLine("Run report:");
            Console.WriteLine("POST " + requestUrl);
            Console.WriteLine(postData + Environment.NewLine);

            // Post to run the report
            var runReportResponse = MakeRequest<ReportUpdate>(requestUrl, WebRequestMethods.Http.Post, out location, postData);
            Console.WriteLine("Generated " + location + Environment.NewLine);
            requestUrl = rootUrl + location;
            Console.WriteLine("Please wait");
            while (runReportResponse.status != "completed")

```

```

        }
        runReportResponse = MakeRequest<ReportUpdate>(requestUrl + ".json", WebRequestMethods.Http.Get, out location);
        Console.WriteLine(string.Format("Percent completed {0}, seconds remaining {1}",runReportResponse.percent,
runReportResponse.remaining_seconds));
        Thread.Sleep(1000);
    }
    Console.WriteLine("Completed"+ Environment.NewLine);

    // Get all queries (In this example it is only one)
    var getQueriesResponse = MakeRequest<List<QueryResult>>(requestUrl + "/queries.json", WebRequestMethods.Http.Get, out
location);
    string columns = jsondata.criteria.query.columns.Select(c=>c.ToString()).Aggregate((i, j) => i + ", " + j);
    // Print the data from all queries
    foreach (var query in getQueriesResponse) {
        var qr = MakeRequest<QueryData>(requestUrl + "/queries/" + query.id + ".json?offset=0&limit=20&columns=" + columns,
            WebRequestMethods.Http.Get, out location);
        string columnList = jsondata.criteria.query.columns.Select(c=>query.columns.Where(col => col.id == c).First().name)
            .Aggregate((l,r) => l + ", " + r);
        Console.WriteLine(columnList);

        foreach (var dr in qr.data)
        {
            Console.WriteLine(dr.Aggregate((i, j) => i + ', ' + j));
        }
    }
    catch (Exception e)
    {
        Console.WriteLine(e.Message);
    }
}

private static string Base64Encode(string toEncode)
{
    byte[] toEncodeAsBytes
    = System.Text.ASCIIEncoding.ASCII.GetBytes(toEncode);
    return System.Convert.ToBase64String(toEncodeAsBytes);
}

/// <summary>
/// Make request
/// </summary>
/// <typeparam name="T">return type</typeparam>
/// <param name="requestUrl">url for request</param>
/// <param name="action">Http Verb, Get, Post etc</param>
/// <param name="location">location returned from response header </param>
/// <param name="requestData">Data posted</param>
/// <returns></returns>
private static T MakeRequest<T>(string requestUrl, string action, out string location, string requestData = null) where T : class
{
    HttpWebRequest request = WebRequest.Create(requestUrl) as HttpWebRequest;
    request.Headers.Add("Authorization: Basic " + Base64Encode(BASIC_AUTH));
    request.ContentType = "application/json";
    request.Method = action;
    if (requestData == null)
    {
        request.ContentLength = 0;
    }
    else
    {
        ASCIIEncoding encoding = new ASCIIEncoding();
        byte[] byte1 = encoding.GetBytes(requestData);
        request.ContentLength = byte1.Length;
        using (Stream newStream = request.GetRequestStream())
        {
            newStream.Write(byte1, 0, byte1.Length);
        }
    }
}

using (HttpWebResponse response = request.GetResponse() as HttpWebResponse)
{
    if (response.StatusCode != HttpStatusCode.OK && response.StatusCode != HttpStatusCode.Created)
        throw new Exception(string.Format(
            "Unable to run report! StatusCode={0}, Description={1}",
            response.StatusCode,
            response.StatusDescription));
    location = response.Headers[HttpResponseHeader.Location];
    DataContractJsonSerializer jsonSerializer = new DataContractJsonSerializer(typeof(T));
    object objResponse = jsonSerializer.ReadObject(response.GetResponseStream());
    return objResponse as T;
}

```

```

    }
  }
}

```

## CascadeRestClient.csproj:

```

<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="4.0" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <Configuration Condition=" '$(Configuration)' == '' ">Debug</Configuration>
    <Platform Condition=" '$(Platform)' == '' ">x86</Platform>
    <ProductVersion>8.0.30703</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>{4ED69347-523B-46AB-B259-47EF60D4F13A}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <AppDesignerFolder>Properties</AppDesignerFolder>
    <RootNamespace>CascadeRestClient</RootNamespace>
    <AssemblyName>CascadeRestClient</AssemblyName>
    <TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
    <TargetFrameworkProfile>
    </TargetFrameworkProfile>
    <FileAlignment>512</FileAlignment>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|x86' ">
    <PlatformTarget>x86</PlatformTarget>
    <DebugSymbols>true</DebugSymbols>
    <DebugType>full</DebugType>
    <Optimize>false</Optimize>
    <OutputPath>bin\Debug</OutputPath>
    <DefineConstants>DEBUG;TRACE</DefineConstants>
    <ErrorReport>prompt</ErrorReport>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|x86' ">
    <PlatformTarget>x86</PlatformTarget>
    <DebugType>pdbonly</DebugType>
    <Optimize>true</Optimize>
    <OutputPath>bin\Release</OutputPath>
    <DefineConstants>TRACE</DefineConstants>
    <ErrorReport>prompt</ErrorReport>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <ItemGroup>
    <Reference Include="System" />
    <Reference Include="System.Core" />
    <Reference Include="System.Runtime.Serialization" />
    <Reference Include="System.Web.Extensions">
      <HintPath>..\..\..\..\..\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\NETFramework\v4.0\System.Web.Extensions.dll</HintPath>
    </Reference>
    <Reference Include="System.Xml.Linq" />
    <Reference Include="System.Data.DataSetExtensions" />
    <Reference Include="Microsoft.CSharp" />
    <Reference Include="System.Data" />
    <Reference Include="System.Xml" />
  </ItemGroup>
  <ItemGroup>
    <Compile Include="Program.cs" />
    <Compile Include="Properties\AssemblyInfo.cs" />
  </ItemGroup>
  <Import Project="$(MSBuildToolsPath)\Microsoft.CSharp.targets" />
  <!-- To modify your build process, add your task inside one of the targets below and uncomment it.
       Other similar extension points exist, see Microsoft.Common.targets.
  <Target Name="BeforeBuild">
  </Target>
  <Target Name="AfterBuild">
  </Target>
  -->
</Project>

```

## Resources

### Protocols: Get protocol

Get information on one protocol.

```
GET https://{device}/api/profiler/1.2/protocols/{proto}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": 6,
    "name": "tcp"
  },
  {
    "id": 17,
    "name": "udp"
  }
]
```

Property Name	Type	Description	Notes
<i>CProtocols</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of Protocols objects.	
<i>CProtocols</i> [CProtocol]	<i>&lt;object&gt;</i>	Object representing Protocol information.	Optional
<i>CProtocols</i> [CProtocol].id	<i>&lt;number&gt;</i>	ID of the Protocol.	Optional
<i>CProtocols</i> [CProtocol].name	<i>&lt;string&gt;</i>	Name of the Protocol.	Optional

## Protocols: List protocols

Get a list of all supported protocols.

```
GET https://{device}/api/profiler/1.2/protocols
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": 6,
    "name": "tcp"
  },
  {
    "id": 17,
    "name": "udp"
  }
]
```

Property Name	Type	Description	Notes
---------------	------	-------------	-------

<i>CProtocols</i>	<array of <object>>	List of Protocols objects.	
<i>CProtocols</i> [CProtocol]	<object>	Object representing Protocol information.	Optional
<i>CProtocols</i> [CProtocol].id	<number>	ID of the Protocol.	Optional
<i>CProtocols</i> [CProtocol].name	<string>	Name of the Protocol.	Optional

## Vnis: List VNIs

Get a list of Virtual Network Identifiers.

```
GET https://{device}/api/profiler/1.2/vnis
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "description": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "description": "Customer A. Blue Network.",
    "name": "Blue_Network",
    "id": 100
  },
  {
    "description": "Customer B. Blue Network.",
    "name": "Red_Network",
    "id": 200
  }
]
```

Property Name	Type	Description	Notes
<i>VNIs</i>	<array of <object>>	List of VNIs (Virtual Network Identifiers of SDN setup).	
<i>VNIs</i> [VNI]	<object>	Object representing a VNI.	Optional
<i>VNIs</i> [VNI].id	<number>	ID of the VNI.	
<i>VNIs</i> [VNI].description	<string>	Description of the VNI.	Optional
<i>VNIs</i> [VNI].name	<string>	Name of the VNI.	Optional

## Vnis: Delete VNI

Delete a Virtual Network Identifier.

```
DELETE https://{device}/api/profiler/1.2/vnis/{vni_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Vnis: Update VNIs

Update one or many Virtual Network Identifiers.

```
PUT https://{device}/api/profiler/1.2/vnis
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
[
  {
    "id": number,
    "description": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "description": "Customer A. Blue Network.",
    "name": "Blue_Network",
    "id": 100
  },
  {
    "description": "Customer B. Blue Network.",
    "name": "Red_Network",
    "id": 200
  }
]
```

Property Name	Type	Description	Notes
<i>VNIs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of VNIs (Virtual Network Identifiers of SDN setup).	
<i>VNIs[VNI]</i>	<i>&lt;object&gt;</i>	Object representing a VNI.	Optional
<i>VNIs[VNI].id</i>	<i>&lt;number&gt;</i>	ID of the VNI.	
<i>VNIs[VNI].description</i>	<i>&lt;string&gt;</i>	Description of the VNI.	Optional
<i>VNIs[VNI].name</i>	<i>&lt;string&gt;</i>	Name of the VNI.	Optional

## Response Body

On success, the server does not provide any body in the responses.

## Vnis: Get VNI

Get a Virtual Network Identifier.

```
GET https://{device}/api/profiler/1.2/vnis/{vni_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "id": number,
  "description": string,
  "name": string
}
```

Example:

```
{
  "description": "Customer A. Blue Network.",
  "name": "Blue_Network",
  "id": 100
}
```

Property Name	Type	Description	Notes
VNI	<object>	Object representing a VNI.	
VNI.id	<number>	ID of the VNI.	
VNI.description	<string>	Description of the VNI.	Optional
VNI.name	<string>	Name of the VNI.	Optional

## Vnis: Update VNI

Update one Virtual Network Identifier.

```
PUT https://{device}/api/profiler/1.2/vnis/{vni_id}
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

#### JSON

```
{
  "id": number,
  "description": string,
  "name": string
}
```

Example:

```
{
  "description": "Customer A. Blue Network.",
  "name": "Blue_Network",
  "id": 100
}
```

Property Name	Type	Description	Notes
VNI	<object>	Object representing a VNI.	
VNI.id	<number>	ID of the VNI.	
VNI.description	<string>	Description of the VNI.	Optional
VNI.name	<string>	Name of the VNI.	Optional

### Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Disable Steelhead polling

Disables data polling from Steelheads.

```
POST https://{device}/api/profiler/1.2/steelheads/sync/disable
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

#### JSON



```
[
  {
    "ipaddr": string
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<i>SteelheadIPAdrs</i>	<array of <object>>	IP addresses object representing the list of Steelheads.	
<i>SteelheadIPAdrs[SteelheadIPAddr]</i>	<object>	IP address collection object representing the list of Steelheads.	Optional
<i>SteelheadIPAdrs[SteelheadIPAddr].ipaddr</i>	<string>	IP address representing a Steelhead.	

## Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Enable Steelhead polling

Enables data polling from Steelheads.

POST https://{device}/api/profiler/1.2/steelheads/sync/enable

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "ipaddr": string
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<i>SteelheadIPAdrs</i>	<array of <object>>	IP addresses object representing the list of Steelheads.	
<i>SteelheadIPAdrs[SteelheadIPAddr]</i>	<object>	IP address collection object representing the list of Steelheads.	Optional
<i>SteelheadIPAdrs[SteelheadIPAddr].ipaddr</i>	<string>	IP address representing a Steelhead.	

## Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Delete Steelheads global OAuth

Deletes global OAuth code.

```
DELETE https://{device}/api/profiler/1.2/steelheads/oauth_code/global
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Get Steelheads OAuth codes

Get a list of steelheads CIDRs with OAuth code configured.

```
GET https://{device}/api/profiler/1.2/steelheads/oauth_code
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "cidr": string
  }
]
```

Example:

```
[
  {
    "cidr": "10.10.0.60"
  },
  {
    "cidr": "0/0"
  },
  {
    "cidr": "10/8"
  }
]
```

Property Name	Type	Description	Notes
<i>Cidrs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of Cidr Objects.	
<i>Cidrs[Cidr]</i>	<i>&lt;object&gt;</i>	Object representing an IP address, CIDR.	Optional
<i>Cidrs[Cidr].cidr</i>	<i>&lt;string&gt;</i>	IP address, CIDR.	

## Steelheads: Update Steelheads OAuth codes

Creates or Updates Oauth Codes for a list of Steelheads.

```
PUT https://{device}/api/profiler/1.2/steelheads/oauth_code
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "code": string,
    "cidr": string
  }
]

Example:
[
  {
    "cidr": "10.20.0.60",
    "code": "code_ip"
  },
  {
    "cidr": "0/0",
    "code": "code_global"
  },
  {
    "cidr": "10/8",
    "code": "code_region"
  }
]
```

Property Name	Type	Description	Notes
<i>Oauthcodes</i>	<array of <object>>	List of OAuth code Objects.	
<i>Oauthcodes</i> [Oauthcode]	<object>	Object representing an IP address, CIDR and its OAuth code.	Optional
<i>Oauthcodes</i> [Oauthcode].code	<string>	OAuth code.	
<i>Oauthcodes</i> [Oauthcode].cidr	<string>	IP address, CIDR.	

## Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Sync Steelheads QoS data

Retrieves QoS data from Steelheads on which polling is enabled.

POST https://{device}/api/profiler/1.2/steelheads/qos/sync

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
[
  {
    "ipaddr": string
  }
]

Example:
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<i>SteelheadIPAddrs</i>	<array of <object>>	IP addresses object representing the list of Steelheads.	
<i>SteelheadIPAddrs</i> [SteelheadIPAddr]	<object>	IP address collection object representing the list of Steelheads.	Optional

<code>SteelheadIPAddr[SteelheadIPAddr].ipaddr</code>	<code>&lt;string&gt;</code>	IP address representing a Steelhead.	
--	-----------------------------	--------------------------------------	--

## Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Sync Steelhead apps

Retrieves application data from Steelheads on which polling is enabled.

POST `https://{device}/api/profiler/1.2/steelheads/apps/sync`

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
[
  {
    "ipaddr": string
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<code>SteelheadIPAddr</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	IP addresses object representing the list of Steelheads.	
<code>SteelheadIPAddr[SteelheadIPAddr]</code>	<code>&lt;object&gt;</code>	IP address collection object representing the list of Steelheads.	Optional
<code>SteelheadIPAddr[SteelheadIPAddr].ipaddr</code>	<code>&lt;string&gt;</code>	IP address representing a Steelhead.	

## Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Get Steelheads

Get a Steelhead QoS Global Configuration by IP address.

GET `https://{device}/api/profiler/1.2/steelheads/{steelhead_ip}`

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```

{
  "dpi": string,
  "marking": string,
  "sync": {
    "apps": {
      "enabled": string,
      "error_text": string,
      "last_sync_ts": number,
      "last_success_ts": number,
      "error_id": number,
      "state": string
    },
    "qos": {
      "enabled": string,
      "error_text": string,
      "last_sync_ts": number,
      "last_success_ts": number,
      "error_id": number,
      "state": string
    }
  },
  "ipaddr": string,
  "name": string,
  "oauth_custom": string,
  "hier_mode": string,
  "shaping": string,
  "easy_mode": string,
  "bw_overcommit": string
}

```

Example:

```

{
  "marking": false,
  "name": "SH-DataCenter",
  "ipaddr": "10.100.100.252",
  "bw_overcommit": false,
  "sync": {
    "qos": {
      "last_sync_ts": 1370967843,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    },
    "apps": {
      "last_sync_ts": 1370967843,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    }
  },
  "shaping": true,
  "hier_mode": true,
  "oauth_custom": false,
  "dpi": true,
  "easy_mode": false
}

```

Property Name	Type	Description	Notes
<i>Steelhead</i>	<object>	Object representing a steelhead.	
<i>Steelhead.dpi</i>	<string>	Flag indicating if Deep Packet Inspection (DPI) is enabled on this Steelhead.	Optional
<i>Steelhead.marking</i>	<string>	Flag indicating if QoS Marking is enabled on this Steelhead.	Optional
<i>Steelhead.sync</i>	<object>	Object representing Steelhead synchronization information.	
<i>Steelhead.sync.apps</i>	<object>	Object representing Steelhead application synchronization information.	
<i>Steelhead.sync.apps.enabled</i>	<string>	Flag - Enable application synchronization.	
<i>Steelhead.sync.apps.error_text</i>	<string>	Error description.	
<i>Steelhead.sync.apps.last_sync_ts</i>	<number>	Last attempted application synchronization time.	
<i>Steelhead.sync.apps.last_success_ts</i>	<number>	Last successful application synchronization time.	
<i>Steelhead.sync.apps.error_id</i>	<number>	Error ID.	

<i>Steelhead.sync.apps.state</i>	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Steelhead.sync.qos</i>	<object>	Object representing Steelhead QoS synchronization information.	
<i>Steelhead.sync.qos.enabled</i>	<string>	Flag indicating if QoS synchronization is enabled on this Steelhead.	
<i>Steelhead.sync.qos.error_text</i>	<string>	Error description.	
<i>Steelhead.sync.qos.last_sync_ts</i>	<number>	Last attempted QoS synchronization time.	
<i>Steelhead.sync.qos.last_success_ts</i>	<number>	Last successful QoS synchronization time.	
<i>Steelhead.sync.qos.error_id</i>	<number>	Error ID.	
<i>Steelhead.sync.qos.state</i>	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Steelhead.ipaddr</i>	<string>	Steelhead IP address.	
<i>Steelhead.name</i>	<string>	Steelhead name.	
<i>Steelhead.oauth_custom</i>	<string>	Flag indicating if Custom OAuth code is configured on this Steelhead.	
<i>Steelhead.hier_mode</i>	<string>	Flag indicating if QoS Hierarchical Mode is enabled on this Steelhead.	Optional
<i>Steelhead.shaping</i>	<string>	Flag indicating if QoS Shaping is enabled on this Steelhead.	Optional
<i>Steelhead.easy_mode</i>	<string>	Flag indicating which QoS Configuration Mode (Basic/Advanced) is set (Basic if true).	Optional
<i>Steelhead.bw_overcommit</i>	<string>	Flag indicating if QoS Bandwidth Overcommit is enabled on this Steelhead.	Optional

## Steelheads: Delete Steelheads OAuth codes

Deletes OAuth codes for a list of Steelheads.

```
DELETE https://{device}/api/profiler/1.2/steelheads/oauth_code
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "cidr": string
  }
]
```

Example:

```
[
  {
    "cidr": "10.10.0.60"
  },
  {
    "cidr": "0/0"
  },
  {
    "cidr": "10/8"
  }
]
```

Property Name	Type	Description	Notes
<i>Cidrs</i>	<array of <object>>	List of Cidr Objects.	
<i>Cidrs[Cidr]</i>	<object>	Object representing an IP address, CIDR.	Optional
<i>Cidrs[Cidr].cidr</i>	<string>	IP address, CIDR.	

### Response Body

On success, the server does not provide any body in the responses.

## Steelheads: Update Steelheads global OAuth

Creates or Updates global OAuth code.

```
PUT https://{device}/api/profiler/1.2/steelheads/oauth_code/global
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

#### JSON

```
{  
  "code": string  
}
```

Example:

```
{  
  "code": "code_global"  
}
```

Property Name	Type	Description	Notes
<i>Globalcode</i>	<object>	Global OAuth code object.	
<i>Globalcode.code</i>	<string>	OAuth Code.	

### Response Body

On success, the server does not provide any body in the responses.

---

## Steelheads: List Steelheads

Get a list of Steelheads and their QoS global and application configuration.

```
GET https://{device}/api/profiler/1.2/steelheads
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
[  
  {  
    "dpi": string,  
    "marking": string,  
    "sync": {  
      "apps": {  
        "enabled": string,  
        "error_text": string,  
        "last_sync_ts": number,  
        "last_success_ts": number,  
        "error_id": number,  
        "state": string  
      },  
      "qos": {  
        "enabled": string,  
        "error_text": string,  
        "last_sync_ts": number,  
        "last_success_ts": number,  
        "error_id": number,  
        "state": string  
      }  
    },  
    "ipaddr": string,  
    "name": string,  
    "oauth_custom": string,  
    "hier_mode": string,  
    "shaping": string,  
    "easv_mode": string  
  }  
]
```

```
    "bw_overcommit": string
  }
]
```

Example:

```
[
{
  "marking": false,
  "name": "10.99.15.252",
  "ipaddr": "10.99.15.252",
  "bw_overcommit": false,
  "sync": {
    "qos": {
      "last_sync_ts": 1370967813,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    },
    "apps": {
      "last_sync_ts": 1370967813,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    }
  },
  "shaping": true,
  "hier_mode": true,
  "oauth_custom": false,
  "dpi": false,
  "easy_mode": true
},
{
  "marking": false,
  "name": "SH-DataCenter",
  "ipaddr": "10.100.100.252",
  "bw_overcommit": false,
  "sync": {
    "qos": {
      "last_sync_ts": 1370967843,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    },
    "apps": {
      "last_sync_ts": 1370967843,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    }
  },
  "shaping": true,
  "hier_mode": true,
  "oauth_custom": false,
  "dpi": true,
  "easy_mode": false
},
{
  "marking": false,
  "name": "SH-LosAngeles",
  "ipaddr": "10.99.12.252",
  "bw_overcommit": false,
  "sync": {
    "qos": {
      "last_sync_ts": 1371583559,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    },
    "apps": {
      "last_sync_ts": 1371583559,
      "enabled": true,
      "last_success_ts": 0,
      "state": "SYNC_FAILED",
      "error_id": 720897,
      "error_text": "28: Timeout was reached"
    }
  }
},
]
```



```

"shaping": true,
"hier_mode": true,
"oauth_custom": false,
"dpi": false,
"easy_mode": true
}
]

```

Property Name	Type	Description	Notes
<i>Steelheads</i>	<array of <object>>	List of Steelheads and their QoS and Application configuration data.	
<i>Steelheads[Steelhead]</i>	<object>	Steelhead QoS and Application Configuration Data.	Optional
<i>Steelheads[Steelhead].dpi</i>	<string>	Flag indicating if Deep Packet Inspection (DPI) is enabled on this Steelhead.	Optional
<i>Steelheads[Steelhead].marking</i>	<string>	Flag indicating if QoS Marking is enabled on this Steelhead.	Optional
<i>Steelheads[Steelhead].sync</i>	<object>	Object representing Steelhead synchronization information.	
<i>Steelheads[Steelhead].sync.apps</i>	<object>	Object representing Steelhead application synchronization information.	
<i>Steelheads[Steelhead].sync.apps.enabled</i>	<string>	Flag - Enable application synchronization.	
<i>Steelheads[Steelhead].sync.apps.error_text</i>	<string>	Error description.	
<i>Steelheads[Steelhead].sync.apps.last_sync_ts</i>	<number>	Last attempted application synchronization time.	
<i>Steelheads[Steelhead].sync.apps.last_success_ts</i>	<number>	Last successful application synchronization time.	
<i>Steelheads[Steelhead].sync.apps.error_id</i>	<number>	Error ID.	
<i>Steelheads[Steelhead].sync.apps.state</i>	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Steelheads[Steelhead].sync.qos</i>	<object>	Object representing Steelhead QoS synchronization information.	
<i>Steelheads[Steelhead].sync.qos.enabled</i>	<string>	Flag indicating if QoS synchronization is enabled on this Steelhead.	
<i>Steelheads[Steelhead].sync.qos.error_text</i>	<string>	Error description.	
<i>Steelheads[Steelhead].sync.qos.last_sync_ts</i>	<number>	Last attempted QoS synchronization time.	
<i>Steelheads[Steelhead].sync.qos.last_success_ts</i>	<number>	Last successful QoS synchronization time.	
<i>Steelheads[Steelhead].sync.qos.error_id</i>	<number>	Error ID.	
<i>Steelheads[Steelhead].sync.qos.state</i>	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Steelheads[Steelhead].ipaddr</i>	<string>	Steelhead IP address.	
<i>Steelheads[Steelhead].name</i>	<string>	Steelhead name.	
<i>Steelheads[Steelhead].oauth_custom</i>	<string>	Flag indicating if Custom OAuth code is configured on this Steelhead.	
<i>Steelheads[Steelhead].hier_mode</i>	<string>	Flag indicating if QoS Hierarchical Mode is enabled on this Steelhead.	Optional
<i>Steelheads[Steelhead].shaping</i>	<string>	Flag indicating if QoS Shaping is enabled on this Steelhead.	Optional
<i>Steelheads[Steelhead].easy_mode</i>	<string>	Flag indicating which QoS Configuration Mode (Basic/Advanced) is set (Basic if true).	Optional
<i>Steelheads[Steelhead].bw_overcommit</i>	<string>	Flag indicating if QoS Bandwidth Overcommit is enabled on this Steelhead.	Optional

## Steelheads: Ping Steelhead

Pings a Steelhead by IP address.

```
GET https://{device}/api/profiler/1.2/steelheads/{steelhead_ip}/ping
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

---

## Steelheads: Check Steelheads global OAuth

Checks if the global OAuth code is configured.

```
GET https://{device}/api/profiler/1.2/steelheads/oauth_code/global
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{  
  "configured": string  
}
```

Example:

```
{  
  "configured": true  
}
```

Property Name	Type	Description	Notes
<i>OAuthcodeglobal</i>	<i>&lt;object&gt;</i>	Object representing the global OAuth code is configured.	
<i>OAuthcodeglobal.configured</i>	<i>&lt;string&gt;</i>	True if the global OAuth code is configured.	

---

## Devices: Enable REST polling

Globally disable REST polling for all devices.

```
POST https://{device}/api/profiler/1.2/devices/restsync/enable
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

---

## Devices: Get device

Get a device by IP Address.

```
GET https://{device}/api/profiler/1.2/devices/{device_ip}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```

{
  "id": number,
  "type_id": number,
  "ipaddr": string,
  "name": string,
  "type": string,
  "version": string
}

```

Example:

```

{
  "name": "MyNetflowDevice",
  "type_id": 2,
  "ipaddr": "10.0.0.1",
  "version": "N/A",
  "type": "Netflow",
  "id": 123
}

```

Property Name	Type	Description	Notes
<i>Device</i>	<object>	Object representing a device.	
<i>Device.id</i>	<number>	Device identifier (ID). Used internally in the product and in the API.	
<i>Device.type_id</i>	<number>	Device type ID; a way to represent device type that is more friendly to programs.	
<i>Device.ipaddr</i>	<string>	Device IP address.	
<i>Device.name</i>	<string>	Device name, which usually comes from SNMP or DNS.	
<i>Device.type</i>	<string>	Device type, e.g. Cascade Gateway, Cascade Shark or Netflow device.	
<i>Device.version</i>	<string>	Version of the protocol used to communicate with the device.	

## Devices: List devices

Get a list of devices.

```
GET https://{device}/api/profiler/1.2/devices?type_id={number}&cidr={string}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>type_id</i>	<number>	Filter devices by device type.	Optional
<i>cidr</i>	<string>	Filter devices by IP or Subnet (e.g. 10.0.0.0/8).	Optional

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "type_id": number,
    "ipaddr": string,
    "name": string,
    "type": string,
    "version": string
  }
]
```

Example:

```
[
  {
    "name": "MyNetflowDevice",
    "type_id": 2,
    "ipaddr": "10.0.0.1",
    "version": "N/A",
    "type": "Netflow",
    "id": 123
  },
  {
    "name": "MySensorDevice",
    "type_id": 1,
    "ipaddr": "10.0.0.2",
    "version": "M8.4",
    "type": "Sensor",
    "id": 124
  }
]
```

Property Name	Type	Description	Notes
<i>Devices</i>	<array of <object>>	List of network devices that report data to Profiler.	
<i>Devices[Device]</i>	<object>	One device from the list of devices that report data.	Optional
<i>Devices[Device].id</i>	<number>	Device identifier (ID). Used internally in the product and in the API.	
<i>Devices[Device].type_id</i>	<number>	Device type ID; a way to represent device type that is more friendly to programs.	
<i>Devices[Device].ipaddr</i>	<string>	Device IP address.	
<i>Devices[Device].name</i>	<string>	Device name, which usually comes from SNMP or DNS.	
<i>Devices[Device].type</i>	<string>	Device type, e.g. Cascade Gateway, Cascade Shark or Netflow device.	
<i>Devices[Device].version</i>	<string>	Version of the protocol used to communicate with the device.	

## Devices: Check REST polling

Get global flag showing whether REST polling is enabled for all devices.

```
GET https://{device}/api/profiler/1.2/devices/restsync
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "enabled": string
}
```

Property Name	Type	Description	Notes
<i>DevicesRestSyncEnabled</i>	<object>	Object representing global REST sync enabled flag for devices on Profiler.	
<i>DevicesRestSyncEnabled.enabled</i>	<string>	Global REST sync enabled flag for devices on Profiler.	

## Devices: Delete device

## DEVICES: DELETE DEVICE

Delete a device by IP Address. Warning: the device will be deleted in a few minutes after this call.

```
DELETE https://{device}/api/profiler/1.2/devices/{device_ip}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

---

## Devices: Disable REST polling

Globally enable REST polling for all devices.

```
POST https://{device}/api/profiler/1.2/devices/restsync/disable
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

---

## Dscps: List DSCPs

Get complete DSCP configuration.

```
GET https://{device}/api/profiler/1.2/dscps
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "description": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "description": "Assured Forwarding Class 1 Medium Drop",
    "name": "AF12",
    "id": 12
  },
  {
    "description": "",
    "name": "",
    "id": 13
  },
  {
    "description": "Assured Forwarding Class 1 High Drop",
    "name": "AF13",
    "id": 14
  }
]
```

Property Name	Type	Description	Notes
---------------	------	-------------	-------

<i>CDSCPDefs</i>	<array of <object>>	List of DSCP objects.	
<i>CDSCPDefs</i> [DSCPDef]	<object>	Object representing DSCP information.	Optional
<i>CDSCPDefs</i> [DSCPDef].id	<number>	ID of the DSCP.	
<i>CDSCPDefs</i> [DSCPDef].description	<string>	Description of the DSCP.	
<i>CDSCPDefs</i> [DSCPDef].name	<string>	Name of the DSCP.	

## Dscps: Get DSCP

Get information about a specific DSCP.

```
GET https://{device}/api/profiler/1.2/dscps/{dscp_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "id": number,
  "description": string,
  "name": string
}
```

Example:

```
{
  "description": "Assured Forwarding Class 1 Low Drop",
  "name": "AF11",
  "id": 10
}
```

Property Name	Type	Description	Notes
<i>DSCPDef</i>	<object>	Object representing DSCP information.	
<i>DSCPDef</i> .id	<number>	ID of the DSCP.	
<i>DSCPDef</i> .description	<string>	Description of the DSCP.	
<i>DSCPDef</i> .name	<string>	Name of the DSCP.	

## Dscps: Update DSCPs

Update DSCP configuration (only name and description can be updated).

```
PUT https://{device}/api/profiler/1.2/dscps
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "id": number,
    "description": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "description": "Assured Forwarding Class 1 Medium Drop",
    "name": "AF12",
    "id": 12
  },
  {
    "description": "",
    "name": "",
    "id": 13
  },
  {
    "description": "Assured Forwarding Class 1 High Drop",
    "name": "AF13",
    "id": 14
  }
]
```

Property Name	Type	Description	Notes
<i>CDSCPDefs</i>	<array of <object>>	List of DSCP objects.	
<i>CDSCPDefs</i> [DSCPDef]	<object>	Object representing DSCP information.	Optional
<i>CDSCPDefs</i> [DSCPDef].id	<number>	ID of the DSCP.	
<i>CDSCPDefs</i> [DSCPDef].description	<string>	Description of the DSCP.	
<i>CDSCPDefs</i> [DSCPDef].name	<string>	Name of the DSCP.	

## Response Body

On success, the server does not provide any body in the responses.

## Dscps: Update DSCP

Update information for a specific DSCP.

```
PUT https://{device}/api/profiler/1.2/dscps/{dscp_id}
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
{
  "id": number,
  "description": string,
  "name": string
}
```

Example:

```
{
  "description": "Assured Forwarding Class 1 Low Drop",
  "name": "AF11",
  "id": 10
}
```

Property Name	Type	Description	Notes
<i>DSCPDef</i>	<object>	Object representing DSCP information.	
<i>DSCPDef</i> .id	<number>	ID of the DSCP.	
<i>DSCPDef</i> .description	<string>	Description of the DSCP.	
<i>DSCPDef</i> .name	<string>	Name of the DSCP.	

## Response Body

On success, the server does not provide any body in the responses.

## Port\_Groups: Get port group

Get one port group.

```
GET https://{device}/api/profiler/1.2/port_groups/{group_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "id": number,
  "definitions": [
    {
      "port": number,
      "proto": number
    }
  ],
  "name": string
}
```

Example:

```
{
  "definitions": [
    {
      "port": 137,
      "proto": 6
    },
    {
      "port": 137,
      "proto": 17
    },
    {
      "port": 138,
      "proto": 6
    },
    {
      "port": 138,
      "proto": 17
    },
    {
      "port": 139,
      "proto": 6
    },
    {
      "port": 139,
      "proto": 17
    }
  ],
  "id": 3,
  "name": "netbios"
}
```

Property Name	Type	Description	Notes
<i>CPortGroupDef</i>	<object>	Object representing port group information.	
<i>CPortGroupDef.id</i>	<number>	ID of the port group. To be used in the API.	Optional
<i>CPortGroupDef.definitions</i>	<array of <object>>	Object with port group's definitions.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort]	<object>	Port associated with port group.	Optional
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].port	<number>	Port associated with port group.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].proto	<number>	Protocol that corresponds to the port of the port group.	
<i>CPortGroupDef.name</i>	<string>	Name of the port group.	



# Port\_Groups: Create port group

Create a new port group.

POST https://{device}/api/profiler/1.2/port\_groups

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
{
  "id": number,
  "definitions": [
    {
      "port": number,
      "proto": number
    }
  ],
  "name": string
}
```

Example:

```
{
  "definitions": [
    {
      "port": 137,
      "proto": 6
    },
    {
      "port": 137,
      "proto": 17
    },
    {
      "port": 138,
      "proto": 6
    },
    {
      "port": 138,
      "proto": 17
    },
    {
      "port": 139,
      "proto": 6
    },
    {
      "port": 139,
      "proto": 17
    }
  ],
  "id": 3,
  "name": "netbios"
}
```

Property Name	Type	Description	Notes
<i>CPortGroupDef</i>	<object>	Object representing port group information.	
<i>CPortGroupDef.id</i>	<number>	ID of the port group. To be used in the API.	Optional
<i>CPortGroupDef.definitions</i>	<array of <object>>	Object with port group's definitions.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort]	<object>	Port associated with port group.	Optional
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].port	<number>	Port associated with port group.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].proto	<number>	Protocol that corresponds to the port of the port group.	
<i>CPortGroupDef.name</i>	<string>	Name of the port group.	

## Response Body

On success, the server returns a response body with the following structure:

JSON

```

{
  "id": number,
  "definitions": [
    {
      "port": number,
      "proto": number
    }
  ],
  "name": string
}

```

Example:

```

{
  "definitions": [
    {
      "port": 137,
      "proto": 6
    },
    {
      "port": 137,
      "proto": 17
    },
    {
      "port": 138,
      "proto": 6
    },
    {
      "port": 138,
      "proto": 17
    },
    {
      "port": 139,
      "proto": 6
    },
    {
      "port": 139,
      "proto": 17
    }
  ],
  "id": 3,
  "name": "netbios"
}

```

Property Name	Type	Description	Notes
<i>CPortGroupDef</i>	<object>	Object representing port group information.	
<i>CPortGroupDef.id</i>	<number>	ID of the port group. To be used in the API.	Optional
<i>CPortGroupDef.definitions</i>	<array of <object>>	Object with port group's definitions.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort]	<object>	Port associated with port group.	Optional
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].port	<number>	Port associated with port group.	
<i>CPortGroupDef.definitions</i> [CPortGroupProtoPort].proto	<number>	Protocol that corresponds to the port of the port group.	
<i>CPortGroupDef.name</i>	<string>	Name of the port group.	

## Port\_Groups: List port groups

Get a list of all configured port groups.

```
GET https://{device}/api/profiler/1.2/port_groups
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "definitions": [
      {
        "port": number,
        "proto": number
      }
    ],
    "name": string
  }
]
```

Example:

```
[
  {
    "definitions": [
      {
        "port": 25,
        "proto": 6
      },
      {
        "port": 995,
        "proto": 6
      },
      {
        "port": 995,
        "proto": 17
      },
      {
        "port": 1109,
        "proto": 6
      }
    ],
    "id": 2,
    "name": "mail"
  },
  {
    "definitions": [
      {
        "port": 137,
        "proto": 6
      },
      {
        "port": 137,
        "proto": 17
      },
      {
        "port": 138,
        "proto": 6
      },
      {
        "port": 138,
        "proto": 17
      },
      {
        "port": 139,
        "proto": 6
      },
      {
        "port": 139,
        "proto": 17
      }
    ],
    "id": 3,
    "name": "netbios"
  }
]
```

Property Name	Type	Description	Notes
<i>CPortGroupDefs</i>	<array of <object>>	List of Port Group objects.	
<i>CPortGroupDefs</i> [CPortGroupDef]	<object>	Object representing port group information.	Optional
<i>CPortGroupDefs</i> [CPortGroupDef].id	<number>	ID of the port group. To be used in the API.	Optional
<i>CPortGroupDefs</i> [CPortGroupDef].definitions	<array of <object>>	Object with port group's definitions.	
<i>CPortGroupDefs</i> [CPortGroupDef].definitions[CPortGroupProtoPort]	<object>	Port associated with port group.	Optional
<i>CPortGroupDefs</i> [CPortGroupDef].definitions[CPortGroupProtoPort].port	<number>	Port associated with port group.	

<i>CPortGroupDef</i> s[CPortGroupDef].definitions[CPortGroupProtoPort].proto	<number>	Protocol that corresponds to the port of the port group.	
<i>CPortGroupDef</i> s[CPortGroupDef].name	<string>	Name of the port group.	

## Port\_Groups: Update port group

Update one port group.

PUT https://{device}/api/profiler/1.2/port\_groups/{group\_id}

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

#### JSON

```
{
  "id": number,
  "definitions": [
    {
      "port": number,
      "proto": number
    }
  ],
  "name": string
}
```

Example:

```
{
  "definitions": [
    {
      "port": 137,
      "proto": 6
    },
    {
      "port": 137,
      "proto": 17
    },
    {
      "port": 138,
      "proto": 6
    },
    {
      "port": 138,
      "proto": 17
    },
    {
      "port": 139,
      "proto": 6
    },
    {
      "port": 139,
      "proto": 17
    }
  ],
  "id": 3,
  "name": "netbios"
}
```

Property Name	Type	Description	Notes
<i>CPortGroupDef</i>	<object>	Object representing port group information.	
<i>CPortGroupDef</i> .id	<number>	ID of the port group. To be used in the API.	Optional
<i>CPortGroupDef</i> .definitions	<array of <object>>	Object with port group's definitions.	
<i>CPortGroupDef</i> .definitions [CPortGroupProtoPort]	<object>	Port associated with port group.	Optional
<i>CPortGroupDef</i> .definitions [CPortGroupProtoPort].port	<number>	Port associated with port group.	
<i>CPortGroupDef</i> .definitions [CPortGroupProtoPort].proto	<number>	Protocol that corresponds to the port of the port group.	
<i>CPortGroupDef</i> .name	<string>	Name of the port group.	

## Response Body

On success, the server does not provide any body in the responses.

---

## Port\_Groups: Delete

Delete one port group.

```
DELETE https://{device}/api/profiler/1.2/port_groups/{group_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server does not provide any body in the responses.

---

## Ping: Ping

Simple test of service availability.

```
GET https://{device}/api/profiler/1.2/ping
```

## Authorization

This request requires authorization.

## Response Body

On success, the server does not provide any body in the responses.

---

## Sharks: Enable Sharks polling

Enables data polling from Sharks.

```
POST https://{device}/api/profiler/1.2/sharks/sync/enable
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "ipaddr": string
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<i>SharkIPAddr</i>	<array of <object>>	IP addresses object representing the list of all Sharks.	
<i>SharkIPAddr</i> [SharkIPAddr]	<object>	IP address collection object representing the list of all Sharks.	Optional

<code>SharkIPAddr[SharkIPAddr].ipaddr</code>	<code>&lt;string&gt;</code>	IP address representing a Shark.	
--	-----------------------------	----------------------------------	--

## Response Body

On success, the server does not provide any body in the responses.

## Sharks: Sync Sharks apps

Retrieves application data from Sharks on which polling is enabled.

POST `https://{device}/api/profiler/1.2/sharks/apps/sync`

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
[
  {
    "ipaddr": string
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<code>SharkIPAddr</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	IP addresses object representing the list of all Sharks.	
<code>SharkIPAddr[SharkIPAddr]</code>	<code>&lt;object&gt;</code>	IP address collection object representing the list of all Sharks.	Optional
<code>SharkIPAddr[SharkIPAddr].ipaddr</code>	<code>&lt;string&gt;</code>	IP address representing a Shark.	

## Response Body

On success, the server does not provide any body in the responses.

## Sharks: Get Shark apps

Get configuration of a Shark by IP address.

GET `https://{device}/api/profiler/1.2/sharks/{shark_ip}`

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```

{
  "sync": {
    "apps": {
      "enabled": string,
      "error_text": string,
      "last_sync_ts": number,
      "last_success_ts": number,
      "error_id": number,
      "state": string
    }
  },
  "ipaddr": string
}

```

Property Name	Type	Description	Notes
<i>Shark</i>	<object>	Object representing a Shark.	
<i>Shark.sync</i>	<object>	Object representing Shark synchronization information.	
<i>Shark.sync.apps</i>	<object>	Object representing Shark application synchronization information.	
<i>Shark.sync.apps.enabled</i>	<string>	Flag indicating if application synchronization is enabled on this Shark.	
<i>Shark.sync.apps.error_text</i>	<string>	Error description.	
<i>Shark.sync.apps.last_sync_ts</i>	<number>	Last attempted application synchronization time.	
<i>Shark.sync.apps.last_success_ts</i>	<number>	Last successful application synchronization time.	
<i>Shark.sync.apps.error_id</i>	<number>	Error ID.	
<i>Shark.sync.apps.state</i>	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Shark.ipaddr</i>	<string>	Shark IP address.	

## Sharks: Ping Shark

Ping a Shark by IP address.

```
GET https://{device}/api/profiler/1.2/sharks/{shark_ip}/ping
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Sharks: Disable Sharks polling

Disables data polling from Sharks.

```
POST https://{device}/api/profiler/1.2/sharks/sync/disable
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "ipaddr": string
  }
]

Example:
[
  {
    "ipaddr": "10.99.16.252"
  },
  {
    "ipaddr": "10.99.15.252"
  },
  {
    "ipaddr": "10.99.14.252"
  }
]
```

Property Name	Type	Description	Notes
<i>SharkIPAddr</i> s	<array of <object>>	IP addresses object representing the list of all Sharks.	
<i>SharkIPAddr</i> s[ <i>SharkIPAddr</i> ]	<object>	IP address collection object representing the list of all Sharks.	Optional
<i>SharkIPAddr</i> s[ <i>SharkIPAddr</i> ].ipaddr	<string>	IP address representing a Shark.	

## Response Body

On success, the server does not provide any body in the responses.

## Sharks: List Sharks

Get a list of Sharks and their application configuration data.

```
GET https://{device}/api/profiler/1.2/sharks
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "sync": {
      "apps": {
        "enabled": string,
        "error_text": string,
        "last_sync_ts": number,
        "last_success_ts": number,
        "error_id": number,
        "state": string
      }
    },
    "ipaddr": string
  }
]
```

Example:  
[]

Property Name	Type	Description	Notes
<i>Sharks</i>	<array of <object>>	List of Sharks and their configuration data.	
<i>Sharks</i> [ <i>Shark</i> ]	<object>	Shark configuration data.	Optional
<i>Sharks</i> [ <i>Shark</i> ].sync	<object>	Object representing Shark synchronization information.	
<i>Sharks</i> [ <i>Shark</i> ].sync.apps	<object>	Object representing Shark application synchronization information.	



<i>Sharks</i> [Shark].sync.apps.enabled	<string>	Flag indicating if application synchronization is enabled on this Shark.	
<i>Sharks</i> [Shark].sync.apps.error_text	<string>	Error description.	
<i>Sharks</i> [Shark].sync.apps.last_sync_ts	<number>	Last attempted application synchronization time.	
<i>Sharks</i> [Shark].sync.apps.last_success_ts	<number>	Last successful application synchronization time.	
<i>Sharks</i> [Shark].sync.apps.error_id	<number>	Error ID.	
<i>Sharks</i> [Shark].sync.apps.state	<string>	Synchronization status.	Values: SYNC_INITIALIZING, SYNC_FAILED, SYNC_SUCCEEDED, SYNC_DISABLED, SYNC_NA
<i>Sharks</i> [Shark].ipaddr	<string>	Shark IP address.	

## Interfaces: Delete interface

Delete one network interface.

```
DELETE https://{device}/api/profiler/1.2/interfaces/{ip:ifindex}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Interfaces: Update interfaces

Update network interfaces (fields that can be update: label, inbound speed, outbound speed).

```
PUT https://{device}/api/profiler/1.2/interfaces
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "user_inbound_speed": number,
    "ipaddr": string,
    "label": string,
    "user_outbound_speed": number,
    "ifindex": number
  }
]
```

Example:

```
[
  {
    "ifindex": 2,
    "user_outbound_speed": 44153724,
    "ipaddr": "10.2.3.5",
    "user_inbound_speed": 140736208929648,
    "label": "hsdgs"
  },
  {
    "ifindex": 3,
    "user_outbound_speed": 44153724,
    "ipaddr": "10.2.3.5",
    "user_inbound_speed": 140736208929648,
    "label": "jhgvas"
  }
]
```

Property Name	Type	Description	Notes
<i>InterfaceUpdateDefs_1_2</i>	<array of <object>>	List of update interfaces.	

<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2]	<object>	object representing update interface.	Optional
<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2]. user_inbound_speed	<number>	update interface's inbound speed declared by the user.	
<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2].ipaddr	<string>	update interface's IP address.	
<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2].label	<string>	update interface's label.	
<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2]. user_outbound_speed	<number>	update interface's outbound speed declared by the user.	
<i>CInterfaceUpdateDef_1_2</i> [CInterfaceUpdateDef_1_2].ifindex	<number>	update interface's index.	

## Response Body

On success, the server does not provide any body in the responses.

## Interfaces: Get interface

Get one network interface.

```
GET https://{device}/api/profiler/1.2/interfaces/{ip:ifindex}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "mac": string,
  "id": number,
  "ifdescr": string,
  "outbound_speed": number,
  "user_inbound_speed": number,
  "ipaddr": string,
  "label": string,
  "user_outbound_speed": number,
  "ifalias": string,
  "inbound_speed": number,
  "ifindex": number
}
```

Example:

```
{
  "ipaddr": "10.2.3.5",
  "ifalias": "5",
  "user_inbound_speed": 140736208929648,
  "inbound_speed": 140736208929120,
  "label": "4",
  "mac": "08:00:2b:01:02:04",
  "ifdescr": "6",
  "ifindex": 2,
  "outbound_speed": 140736208929104,
  "user_outbound_speed": 44153724,
  "id": 2
}
```

Property Name	Type	Description	Notes
<i>CInterfaceDef_1_2</i>	<object>	Object representing an interface.	
<i>CInterfaceDef_1_2</i> .mac	<string>	Interface's mac address.	
<i>CInterfaceDef_1_2</i> .id	<number>	Interface's ID.	
<i>CInterfaceDef_1_2</i> .ifdescr	<string>	Name (ifDescr).	
<i>CInterfaceDef_1_2</i> .outbound_speed	<number>	Interface's reported outbound speed.	
<i>CInterfaceDef_1_2</i> .user_inbound_speed	<number>	Interface's inbound speed declared by the user.	
<i>CInterfaceDef_1_2</i> .ipaddr	<string>	IP address of the interface.	
<i>CInterfaceDef_1_2</i> .label	<string>	Interface's label.	
<i>CInterfaceDef_1_2</i> .user_outbound_speed	<number>	Interface's outbound speed declared by the user.	

<i>CInterfaceDef_1_2.ifalias</i>	<string>	Description (ifAlias).	
<i>CInterfaceDef_1_2.inbound_speed</i>	<number>	Interface's reported inbound speed.	
<i>CInterfaceDef_1_2.ifindex</i>	<number>	Interface's index.	

## Interfaces: List interfaces

Get a list of all known network interfaces.

```
GET https://{device}/api/profiler/1.2/interfaces?offset={number}&ipaddr={string}&limit={number}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<number>	Starting element number.	Optional
<i>ipaddr</i>	<string>	Filter network interfaces by an IP address.	Optional
<i>limit</i>	<number>	Number of rows to be returned.	Optional

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
[
  {
    "mac": string,
    "id": number,
    "ifdescr": string,
    "outbound_speed": number,
    "user_inbound_speed": number,
    "ipaddr": string,
    "label": string,
    "user_outbound_speed": number,
    "ifalias": string,
    "inbound_speed": number,
    "ifindex": number
  }
]
```

Example:

```
[
  {
    "ipaddr": "10.2.5.5",
    "ifalias": "5",
    "user_inbound_speed": 140736208929648,
    "inbound_speed": 140736208929120,
    "label": "4",
    "mac": "08:00:2b:01:02:04",
    "ifdescr": "6",
    "ifindex": 2,
    "outbound_speed": 140736208929104,
    "user_outbound_speed": 44153724,
    "id": 3
  },
  {
    "ipaddr": "10.2.5.5",
    "ifalias": "5",
    "user_inbound_speed": 140736208929648,
    "inbound_speed": 140736208929120,
    "label": "unique",
    "mac": "08:00:2b:01:02:05",
    "ifdescr": "6",
    "ifindex": 2,
    "outbound_speed": 140736208929104,
    "user_outbound_speed": 44153724,
    "id": 4
  }
]
```

Property Name	Type	Description	Notes
<i>CInterfaceDefs_1_2</i>	<array of <object>>	List of interfaces.	

<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2]	<object>	Object representing an interface.	Optional
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].mac	<string>	Interface's mac address.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].id	<number>	Interface's ID.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].ifdescr	<string>	Name (ifDescr).	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].outbound_speed	<number>	Interface's reported outbound speed.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].user_inbound_speed	<number>	Interface's inbound speed declared by the user.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].ipaddr	<string>	IP address of the interface.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].label	<string>	Interface's label.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].user_outbound_speed	<number>	Interface's outbound speed declared by the user.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].ifalias	<string>	Description (ifAlias).	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].inbound_speed	<number>	Interface's reported inbound speed.	
<i>CInterfaceDefs_1_2</i> [CInterfaceDef_1_2].ifindex	<number>	Interface's index.	

## Reporting: List reports

Get a list of reports with their respective status.

```
GET https://{device}/api/profiler/1.2/reporting/reports
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "run_time": number,
    "error_text": string,
    "remaining_seconds": number,
    "saved": string,
    "id": number,
    "status": string,
    "percent": number,
    "user_id": number,
    "size": number,
    "name": string,
    "template_id": number
  }
]
```

Example:

```
[
  {
    "status": "completed",
    "user_id": 1,
    "name": "",
    "percent": 100,
    "id": 1000,
    "remaining_seconds": 0,
    "run_time": 1352494550,
    "saved": false,
    "template_id": 952,
    "error_text": "",
    "size": 140
  },
  {
    "status": "completed",
    "user_id": 1,
    "name": "Host Information Report",
    "percent": 100,
    "id": 1001,
    "remaining_seconds": 0,
    "run_time": 1352494550,
    "saved": true,
    "template_id": 952,
    "error_text": "",
    "size": 140
  }
]
```

Property Name	Type	Description	Notes
<i>ReportInfoList</i>	<array of <object>>	List of report objects.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ]	<object>	Object representing report information.	Optional
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].run_time	<number>	Time when the report was run (Unix time).	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].error_text	<string>	A report can be completed with an error. Error message may provide more detailed info.	Optional
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].remaining_seconds	<number>	Number of seconds remaining to run the report. Even if this number is 0, the report may not yet be completed, so check 'status' to make sure what the status is.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].saved	<string>	Boolean flag indicating if the report was saved.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].id	<number>	ID of the report. To be used in the API.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].status	<string>	Status of the report.	Values: completed, running, waiting
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].percent	<number>	Progress of the report represented by percentage of report completion.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].user_id	<number>	ID of the user who owns the report.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].size	<number>	Size of the report in kilobytes.	
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].name	<string>	Name of the report. Could be given by a user or automatically generated by the system.	Optional
<i>ReportInfoList</i> [ <i>ReportInfo</i> ].template_id	<number>	ID of the template that the report is based on.	

## Reporting: Get widget

Get one widget from the template section.

GET [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/sections/{section\\_id}/widgets/{widget\\_id}](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets/{widget_id})

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "dscp_app_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        },
        "dscp": {
          "name": string,
          "code_point": number
        }
      }
    ],
    "services": [
      {
        "name": string,
        "service_id": number
      }
    ],
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "comparison_time_frame": {
      "data_resolution": string,
      "refresh_interval": string,
      "type": string
    },
    "host_group_pairs": [
      {
        "server": {
          "name": string,
          "group_id": number
        },
        "client": {
          "name": string,
          "group_id": number
        }
      }
    ],
    "wan_group": string,
    "traffic_expression": string,
    "split_direction": string,
    "include_successes": string,
    "include_non_optimized_sites": string,
    "columns": [
      number
    ],
    "application_servers": [
      {
        "app": {
          "code": string,
          "name": string,
```

```
"tunneled": string
},
"server": {
  "mac": string,
  "ipaddr": string,
  "name": string
}
},
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_comments": string
```

```
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number
```



```

    id: number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}

```

Property Name	Type	Description	Notes
<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional

<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].app	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i> [CDSCPAppPort].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services</i> [CService]	<object>	One CService object.	Optional
<i>TMWidget.criteria.services</i> [CService]. name	<string>	Service name.	
<i>TMWidget.criteria.services</i> [CService]. service_id	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups</i> [CPortGroup]	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups</i> [CPortGroup].name	<string>	Name of the port group.	Optional
<i>TMWidget.criteria.port_groups</i> [CPortGroup].group_id	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame</i> . data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame</i> . refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame</i> . type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair]	<object>	One CHostGroupPair object.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', 'WAN/Optimized', 'WAN/Non-optimized'.	Optional
<i>TMWidget.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMWidget.criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidget.criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMWidget.criteria</i> . include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional

<i>TMWidget.criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMWidget.criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>TMWidget.criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMWidget.criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>TMWidget.criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>TMWidget.criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>TMWidget.criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.application_ports [CApplicationPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.application_ports [CApplicationPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort]</i>	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].server</i>	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports [CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional

<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame</i> . data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> . refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional
<i>TMWidget.criteria.service.name</i>	<string>	Service name.	
<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies</i> [item]	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations</i> [CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations</i> [CServiceLocation].name	<string>	Service location name.	
<i>TMWidget.criteria.service_locations</i> [CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location</i> . location_id	<string>	Service location ID.	Optional
<i>TMWidget.criteria</i> . include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port	<object>	Port specification.	

<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users</i> [CUser]	<object>	One CUser object.	Optional
<i>TMWidget.criteria.users</i> [CUser].name	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	

<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client	<object>	Specification of the client host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol]	<object>	Object representing Protocol information.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol]. id	<number>	ID of the Protocol.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol]. name	<string>	Name of the Protocol.	Optional
<i>TMWidget.criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMWidget.criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget.criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface]	<object>	One CInterface object.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface]. ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface]. name	<string>	Interface name.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface]. ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup]	<object>	One CHostGroup object.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup].name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup].group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional



<i>TMWidget.criteria.dscps</i> [CDSCP]	<object>	One CDSCP object.	Optional
<i>TMWidget.criteria.dscps</i> [CDSCP].name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscps</i> [CDSCP].code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMWidget.criteria.applications</i> [CApplication]	<object>	One CApplication object.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].code	<string>	Application code.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].name	<string>	Application name.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.title</i>	<string>	Widget title.	
<i>TMWidget.attributes</i>	<object>	Widget common attributes.	Optional
<i>TMWidget.attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.attributes.open_nodes</i> [item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<object>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes</i> [item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<number>	Widget height.	Optional



<i>TMWidget</i> .user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget</i> .user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMWidget</i> .user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget</i> .user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget</i> .user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget</i> .user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidget</i> .user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMWidget</i> .user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidget</i> .user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget</i> .user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget</i> .user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget</i> .user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidget</i> .timestamp	<string>	Widget time stamp specification.	Optional

## Reporting: Get factory widget

Get configuration of a stock widget.

```
GET https://{device}/api/profiler/1.2/reporting/templates/widgets/{widget_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "dscp_app_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        },
        "dscp": {
          "name": string,
          "code_point": number
        }
      }
    ],
    "services": [
      {
        "name": string,
        "service_id": number
      }
    ],
    "port_groups": [
      {
        "name": string,
```

```
"group_id": number
}
],
"comparison_time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
```

```
    "ifindex": number
  },
  "dscp": {
    "name": string,
    "code_point": number
  }
},
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
```

```
    "name": string,
    "ifindex": number
  },
  "dst": {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
},
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string
}
```

```

    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "user_attributes": {
    "pan_zoomable": string,
    "line_scale": string,
    "format_bytes": string,
    "show_images": string,
    "open_nodes": [
      string
    ],
    "line_style": string,
    "layout": string,
    "width": number,
    "height": number,
    "percent_of_total": string,
    "edge_thickness": string,
    "display_host_group_type": string,
    "extend_to_zero": string,
    "collapsible": string,
    "high_threshold": string,
    "n_items": number,
    "colspan": number,
    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}

```

Property Name	Type	Description	Notes
<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE

<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMWidget.criteria.services[CService].name</i>	<string>	Service name.	
<i>TMWidget.criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month



<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app	<object>	Application specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame</i> . data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> . refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> .type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional
<i>TMWidget.criteria.service</i> .name	<string>	Service name.	



<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation].name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_locations [CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users[CUser]</i>	<object>	One CUser object.	Optional
<i>TMWidget.criteria.users[CUser].name</i>	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional

<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client	<object>	Specification of the client host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.name	<string>	Host name.	Optional

<i>TMWidget.criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMWidget.criteria.protocols[CProtocol]</i>	<object>	Object representing Protocol information.	Optional
<i>TMWidget.criteria.protocols[CProtocol].id</i>	<number>	ID of the Protocol.	Optional
<i>TMWidget.criteria.protocols[CProtocol].name</i>	<string>	Name of the Protocol.	Optional
<i>TMWidget.criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMWidget.criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget.criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget.criteria.interfaces[CInterface]</i>	<object>	One CInterface object.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.interfaces[CInterface].name</i>	<string>	Interface name.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ifindex</i>	<number>	Interface index.	Optional
<i>TMWidget.criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup]</i>	<object>	One CHostGroup object.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidget.criteria.dscps[CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>TMWidget.criteria.dscps[CDSCP].name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscps[CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMWidget.criteria.applications [CAApplication]</i>	<object>	One CAApplication object.	Optional
<i>TMWidget.criteria.applications [CAApplication].code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.applications [CAApplication].name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.applications [CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.title</i>	<string>	Widget title.	
<i>TMWidget.attributes</i>	<object>	Widget common attributes.	Optional
<i>TMWidget.attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.attributes.format_bytes</i>	<string>	What unit to use for formating traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional

<i>TMWidget.attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes [item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidget.user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidget.user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional

## Reporting: List directions

Get a list of directions that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/directions
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "inn",
    "name": "in"
  },
  {
    "id": "out",
    "name": "out"
  },
  {
    "id": "c2s",
    "name": "client to server"
  },
  {
    "id": "s2c",
    "name": "server to client"
  }
]
```

Property Name	Type	Description	Notes
<i>Directions</i>	<array of <object>>	List of directions.	
<i>Directions</i> [Direction]	<object>	Object representing a direction.	Optional
<i>Directions</i> [Direction].id	<string>	ID of a direction. To be used in the API.	
<i>Directions</i> [Direction].name	<string>	Human-readable name of a direction.	

## Reporting: Update template

Update reporting template.

PUT [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}](https://{device}/api/profiler/1.2/reporting/templates/{template_id})

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
{
  "traffic_expression": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "layout": [
    {
      "flow_items": [
        {
          "id": number
        }
      ],
      "attributes": {
        "wrappable": string,
        "full_width": string,
        "item_spacing": string
      }
    }
  ],
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "last added section id": number
}
```

```
last_added_section_id": number,
"name": string,
"last_added_widget_id": number,
"version": string,
"disabled": string,
"timestamp": string,
"sections": [
  {
    "widgets": [
      {
        "config": {
          "datasource": string,
          "visualization": string,
          "widget_type": string
        },
        "widget_id": number,
        "criteria": {
          "ports": [
            {
              "port": number,
              "protocol": number,
              "name": string
            }
          ],
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,
                "name": string
              },
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              },
              "dscp": {
                "name": string,
                "code_point": number
              }
            }
          ],
          "services": [
            {
              "name": string,
              "service_id": number
            }
          ],
          "port_groups": [
            {
              "name": string,
              "group_id": number
            }
          ],
          "comparison_time_frame": {
            "data_resolution": string,
            "refresh_interval": string,
            "type": string
          },
          "host_group_pairs": [
            {
              "server": {
                "name": string,
                "group_id": number
              },
              "client": {
                "name": string,
                "group_id": number
              }
            }
          ],
          "wan_group": string,
          "traffic_expression": string,
          "split_direction": string,
          "include_successes": string,
          "include_non_optimized_sites": string,
          "columns": [
            number
          ],
          "application_servers": [
            {
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              },
              "server": {
                "mac": string,
```

```
    "ipaddr": string,
    "name": string
  }
},
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
```

```
"port": {
  "port": number,
  "protocol": number,
  "name": string
},
"app": {
  "code": string,
  "name": string,
  "tunneled": string
},
"server": {
  "mac": string,
  "ipaddr": string,
  "name": string
},
"client": {
  "mac": string,
  "ipaddr": string,
  "name": string
}
},
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
```



```
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {

```

```

    {
      "id": number
    }
  ],
  "attributes": {
    "wrappable": string,
    "full_width": string,
    "item_spacing": string
  }
}
]
}
],
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
}

```

Example:

```

{
  "layout": [
    {
      "flow_items": [
        {
          "id": 1
        }
      ]
    }
  ],
  "name": "VOIP - Call Quality and Usage",
  "user_id": 1,
  "timestamp": "1383141976.674345",
  "live": true,
  "last_added_widget_id": 6,
  "traffic_expression": "app VoIP-RTP",
  "version": "1.1",
  "shared": "Private",
  "sections": [
    {
      "widgets": [
        {
          "title": "VoIP-RTP: Applications",
          "timestamp": "1383141976.674383",
          "criteria": {
            "sort_column": 33,
            "traffic_expression": "",
            "sort_desc": true,
            "centricity": "host",
            "limit": 100,
            "columns": [
              17,
              33,
              34,
              757,
              766,
              781,
              803
            ]
          },
          "time_frame": {
            "data_resolution": "15mins",
            "type": "last_hour",
            "refresh_interval": "15mins"
          }
        },
        {
          "attributes": {
            "format_bytes": "UI_PREF",
            "colspan": 2,
            "n_items": 20
          },
          "config": {
            "widget_type": "APPS",
            "visualization": "TABLE",
            "datasource": "TRAFFIC"
          },
          "widget_id": 1
        },
        {
          "title": "VoIP-RTP: Traffic Quality",
          "timestamp": "1383141976.674428",
          "criteria": {
            "traffic_expression": "",
            "sort_desc": true,
            "centricity": "host"
          }
        }
      ]
    }
  ]
}

```

```
centricity": "host",
"columns": [
  803
],
"time_frame": {
  "data_resolution": "min",
  "type": "last_hour",
  "refresh_interval": "min"
}
},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 1,
  "extend_to_zero": false,
  "line_scale": "LINEAR",
  "line_style": "STACKED"
},
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 2
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674459",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      781
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
```

```
"centricity": "host",
"columns": [
  33
],
"time_frame": {
  "data_resolution": "15mins",
  "type": "last_day",
  "refresh_interval": "15mins"
}
},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 2,
  "extend_to_zero": false,
  "line_style": "STACKED"
},
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "host_group_type": "ByLocation",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  }
},
"attributes": {
  "format_bytes": "UI_PREF",
  "show_images": true,
  "layout": "HORIZONTAL_TREE",
  "colspan": 2,
  "moveable_nodes": true,
  "height": 400,
  "edge_thickness": true,
  "pan_zoomable": true,
  "n_items": 10
},
"config": {
  "widget_type": "HOST_GROUP_PAIRS",
  "visualization": "CONN_GRAPH",
  "datasource": "TRAFFIC"
},
"widget_id": 6
}
],
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      },
      {
        "id": 3
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 4
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 5
      }
    ]
  }
]
```

```

    }
  ],
  {
    "flow_items": [
      {
        "id": 6
      }
    ]
  },
  ],
  "section_id": 1
}
],
"id": 5217,
"description": ""
}

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpec</i>	<object>	Reporting template specification object.	
<i>ReportTemplateSpec.traffic_expression</i>	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpec.id</i>	<number>	ID of the report template.	Optional
<i>ReportTemplateSpec.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpec.sharing</i>	<object>	List of the users the template is shared with (see <i>ReportTemplateSharing</i> ).	Optional
<i>ReportTemplateSpec.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpec.sharing.users[item]</i>	<number>	User ID.	Optional
<i>ReportTemplateSpec.layout</i>	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.description</i>	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpec.user_id</i>	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpec.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpec.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpec.last_added_section_id</i>	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpec.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplateSpec.last_added_widget_id</i>	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpec.version</i>	<string>	Version of the specification.	Optional
<i>ReportTemplateSpec.disabled</i>	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpec.timestamp</i>	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpec.sections</i>	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpec.sections[TMSection]</i>	<object>	One <i>TMSection</i> object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget]</i>	<object>	One <i>TMWidget</i> object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService]</i>	<object>	One CService object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', 'WAN/Optimized', 'WAN/Non-optimized'.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server</i>	<object>	Server specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.port</i>	<number>	Port specification.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort]</i>	<object>	One CHostPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<string>	Host name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface]</i>	<object>	One CDSCPIInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].interface</i>	<object>	Interface specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].interface.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].interface.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].interface.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPIInterface].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service</i>	<object>	Watched service.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<string>	Service location name.	

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location</i>	<object>	Watched service location.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<string>	Host name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users[CUser]</i>	<object>	One CUser object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users[CUser].name</i>	<string>	Active Directory user name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort]</i>	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports[CHostGroupPairPort].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<object>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<object>	Segment source.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<object>	Segment destination.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]</i>	<object>	One CHost object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair]</i>	<object>	One CHostPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server</i>	<object>	Specification of the server host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client</i>	<object>	Specification of the client host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.name</i>	<string>	Host name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched protocols.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol]</i>	<i>&lt;object&gt;</i>	Object representing Protocol information.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].id</i>	<i>&lt;number&gt;</i>	ID of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].name</i>	<i>&lt;string&gt;</i>	Name of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.centricity</i>	<i>&lt;string&gt;</i>	Centricity used to run the report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.limit</i>	<i>&lt;number&gt;</i>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface]</i>	<i>&lt;object&gt;</i>	One CInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup]</i>	<i>&lt;object&gt;</i>	One CHostGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched DSCPs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP]</i>	<i>&lt;object&gt;</i>	One CDSCP object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].name</i>	<i>&lt;string&gt;</i>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].code_point</i>	<i>&lt;number&gt;</i>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched applications.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication]</i>	<i>&lt;object&gt;</i>	One CAApplication object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].code</i>	<i>&lt;string&gt;</i>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].name</i>	<i>&lt;string&gt;</i>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].tunneled</i>	<i>&lt;string&gt;</i>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].title</i>	<i>&lt;string&gt;</i>	Widget title.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes</i>	<i>&lt;object&gt;</i>	Widget common attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].section_id</i>	<i>&lt;number&gt;</i>	Section ID.	
<i>ReportTemplateSpec.sections[TMSection].layout</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Internal section layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine]</i>	<i>&lt;object&gt;</i>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of line items.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items [TMFlowItem]</i>	<i>&lt;object&gt;</i>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items [TMFlowItem].id</i>	<i>&lt;number&gt;</i>	Widget ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes</i>	<i>&lt;object&gt;</i>	List of line attributes.	Optional



<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.wrappable</i>	<i>&lt;string&gt;</i>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.full_width</i>	<i>&lt;string&gt;</i>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.item_spacing</i>	<i>&lt;string&gt;</i>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.img</i>	<i>&lt;object&gt;</i>	Images associated with the template.	Optional
<i>ReportTemplateSpec.img.thumbnail</i>	<i>&lt;object&gt;</i>	A thumbnail-size image for the report template.	Optional
<i>ReportTemplateSpec.img.thumbnail.src</i>	<i>&lt;string&gt;</i>	Relative URL of an image.	
<i>ReportTemplateSpec.img.full</i>	<i>&lt;object&gt;</i>	A full-size image for the report template.	Optional
<i>ReportTemplateSpec.img.full.src</i>	<i>&lt;string&gt;</i>	Relative URL of an image.	

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "traffic_expression": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "layout": [
    {
      "flow_items": [
        {
          "id": number
        }
      ],
      "attributes": {
        "wrappable": string,
        "full_width": string,
        "item_spacing": string
      }
    }
  ],
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "last_added_section_id": number,
  "name": string,
  "last_added_widget_id": number,
  "version": string,
  "disabled": string,
  "timestamp": string,
  "sections": [
    {
      "widgets": [
        {
          "config": {
            "datasource": string,
            "visualization": string,
            "widget_type": string
          },
          "widget_id": number,
          "criteria": {
            "ports": [
              {
                "port": number,
                "protocol": number,
                "name": string
              }
            ]
          },
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,
                "name": string
              },
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              }
            }
          ]
        }
      ]
    }
  ]
}
```

```

    tunnelled: string
  },
  "dscp": {
    "name": string,
    "code_point": number
  }
},
"services": [
  {
    "name": string,
    "service_id": number
  }
],
"port_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"comparison_time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunnelled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunnelled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {

```

```
"mac": string,
"ipaddr": string,
"name": string
},
"client": {
  "mac": string,
  "ipaddr": string,
  "name": string
}
},
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
```

```
    "name": string
  },
  "server": {
    "name": string,
    "group_id": number
  },
  "client": {
    "name": string,
    "group_id": number
  }
},
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
```

```

"show_images": string,
"open_nodes": [
  string
],
"line_style": string,
"layout": string,
"width": number,
"height": number,
"percent_of_total": string,
"edge_thickness": string,
"display_host_group_type": string,
"extend_to_zero": string,
"collapsible": string,
"high_threshold": string,
"n_items": number,
"colspan": number,
"low_threshold": string,
"moveable_nodes": string,
"orientation": string,
"modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
}
},
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
}

```

Example:

```

{
  "layout": [
    {
      "flow_items": [
        {
          "id": 1
        }
      ]
    }
  ],
  "name": "VOIP - Call Quality and Usage"
}

```

```
name : VoIP - Call Quality and Usage ,
"user_id": 1,
"timestamp": "1383141976.674345",
"live": true,
"last_added_widget_id": 6,
"traffic_expression": "app VoIP-RTP",
"version": "1.1",
"shared": "Private",
"sections": [
  {
    "widgets": [
      {
        "title": "VoIP-RTP: Applications",
        "timestamp": "1383141976.674383",
        "criteria": {
          "sort_column": 33,
          "traffic_expression": "",
          "sort_desc": true,
          "centricity": "host",
          "limit": 100,
          "columns": [
            17,
            33,
            34,
            757,
            766,
            781,
            803
          ],
          "time_frame": {
            "data_resolution": "15mins",
            "type": "last_hour",
            "refresh_interval": "15mins"
          }
        },
        "attributes": {
          "format_bytes": "UI_PREF",
          "colspan": 2,
          "n_items": 20
        },
        "config": {
          "widget_type": "APPS",
          "visualization": "TABLE",
          "datasource": "TRAFFIC"
        },
        "widget_id": 1
      },
      {
        "title": "VoIP-RTP: Traffic Quality",
        "timestamp": "1383141976.674428",
        "criteria": {
          "traffic_expression": "",
          "sort_desc": true,
          "centricity": "host",
          "columns": [
            803
          ],
          "time_frame": {
            "data_resolution": "min",
            "type": "last_hour",
            "refresh_interval": "min"
          }
        },
        "attributes": {
          "format_bytes": "UI_PREF",
          "colspan": 1,
          "extend_to_zero": false,
          "line_scale": "LINEAR",
          "line_style": "STACKED"
        },
        "config": {
          "widget_type": "TRAFFIC_OVERALL",
          "visualization": "LINE",
          "datasource": "TRAFFIC"
        },
        "widget_id": 2
      },
      {
        "title": "VoIP-RTP: Traffic Quality",
        "timestamp": "1383141976.674459",
        "criteria": {
          "traffic_expression": "",
          "sort_desc": true,
          "centricity": "host",
          "columns": [
            781
          ],
          "time frame": {
```

```
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      33
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_day",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "host_group_type": "ByLocation",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "time frame": {
```

```

    "data_resolution": "15mins",
    "type": "last_hour",
    "refresh_interval": "15mins"
  }
},
"attributes": {
  "format_bytes": "UI_PREF",
  "show_images": true,
  "layout": "HORIZONTAL_TREE",
  "colspan": 2,
  "moveable_nodes": true,
  "height": 400,
  "edge_thickness": true,
  "pan_zoomable": true,
  "n_items": 10
},
"config": {
  "widget_type": "HOST_GROUP_PAIRS",
  "visualization": "CONN_GRAPH",
  "datasource": "TRAFFIC"
},
"widget_id": 6
}
],
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      },
      {
        "id": 3
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 4
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 5
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 6
      }
    ]
  }
],
"section_id": 1
}
],
"id": 5217,
"description": ""
}

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpec</i>	<object>	Reporting template specification object.	
<i>ReportTemplateSpec.traffic_expression</i>	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpec.id</i>	<number>	ID of the report template.	Optional
<i>ReportTemplateSpec.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpec.sharing</i>	<object>	List of the users the template is shared with (see <i>ReportTemplateSharing</i> ).	Optional
<i>ReportTemplateSpec.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpec.sharing.users[item]</i>	<number>	User ID.	Optional



<i>ReportTemplateSpec.layout</i>	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.description</i>	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpec.user_id</i>	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpec.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpec.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpec.last_added_section_id</i>	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpec.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplateSpec.last_added_widget_id</i>	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpec.version</i>	<string>	Version of the specification.	Optional
<i>ReportTemplateSpec.disabled</i>	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpec.timestamp</i>	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpec.sections</i>	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpec.sections[TMSection]</i>	<object>	One TMSection object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget]</i>	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort]</i>	<object>	One CDSCAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService].name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', 'WAN/Optimized', 'WAN/Non-optimized'.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.name</i>	<string>	Application name.	Optional

<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server	<object>	Server specification.	
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.devices [CDevice]	<object>	One CDevice object.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].ipaddr	<string>	Device IP address.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].name	<string>	Device name.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort]	<object>	One CAApplicationPort object.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port	<object>	Port specification.	
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app	<object>	Application specification.	
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpec</i> .sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort]	<object>	One CHostPairPort object.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface]</i>	<object>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface</i>	<object>	Interface specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point</i>	<number>	DSCP code point.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service</i>	<object>	Watched service.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location</i>	<object>	Watched service location.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<i>&lt;number&gt;</i>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<i>&lt;string&gt;</i>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<i>&lt;object&gt;</i>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<i>&lt;string&gt;</i>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<i>&lt;string&gt;</i>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<i>&lt;string&gt;</i>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<i>&lt;object&gt;</i>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<i>&lt;object&gt;</i>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched users.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser]</i>	<i>&lt;object&gt;</i>	One CUser object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser].name</i>	<i>&lt;string&gt;</i>	Active Directory user name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_desc</i>	<i>&lt;string&gt;</i>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_column</i>	<i>&lt;number&gt;</i>	Sorting column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]</i>	<i>&lt;object&gt;</i>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port</i>	<i>&lt;object&gt;</i>	Port specification.	

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port</i>	<i>&lt;number&gt;</i>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol</i>	<i>&lt;number&gt;</i>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name</i>	<i>&lt;string&gt;</i>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server</i>	<i>&lt;object&gt;</i>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client</i>	<i>&lt;object&gt;</i>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched network segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<i>&lt;object&gt;</i>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<i>&lt;object&gt;</i>	Segment source.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<i>&lt;object&gt;</i>	Segment destination.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]</i>	<i>&lt;object&gt;</i>	One CHost object.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair]</i>	<i>&lt;object&gt;</i>	One CHostPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server</i>	<i>&lt;object&gt;</i>	Specification of the server host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client</i>	<i>&lt;object&gt;</i>	Specification of the client host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched protocols.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol]</i>	<i>&lt;object&gt;</i>	Object representing Protocol information.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].id</i>	<i>&lt;number&gt;</i>	ID of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].name</i>	<i>&lt;string&gt;</i>	Name of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.centricity</i>	<i>&lt;string&gt;</i>	Centricity used to run the report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.limit</i>	<i>&lt;number&gt;</i>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface]</i>	<i>&lt;object&gt;</i>	One CInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup]</i>	<i>&lt;object&gt;</i>	One CHostGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [HostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication]</i>	<object>	One CApplication object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].title</i>	<string>	Widget title.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes</i>	<object>	Widget common attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.width</i>	<number>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.height</i>	<number>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional

<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.n_items</code>	<code>&lt;number&gt;</code>	Maximum number of items shown.	Optional
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.colspan</code>	<code>&lt;number&gt;</code>	How many columns the widget occupies in layout.	Optional
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.low_threshold</code>	<code>&lt;string&gt;</code>	Low threshold on the chart (in bytes).	Optional
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes</code>	<code>&lt;string&gt;</code>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.orientation</code>	<code>&lt;string&gt;</code>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.modal_links</code>	<code>&lt;number&gt;</code>	Flag adding modal links on a widget.	Optional
<code>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].timestamp</code>	<code>&lt;string&gt;</code>	Widget time stamp specification.	Optional
<code>ReportTemplateSpec.sections[TMSection].section_id</code>	<code>&lt;number&gt;</code>	Section ID.	
<code>ReportTemplateSpec.sections[TMSection].layout</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Internal section layout.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine]</code>	<code>&lt;object&gt;</code>	One horizontal line of widgets.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	List of line items.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem]</code>	<code>&lt;object&gt;</code>	Object representing one layout item.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id</code>	<code>&lt;number&gt;</code>	Widget ID.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes</code>	<code>&lt;object&gt;</code>	List of line attributes.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.wrappable</code>	<code>&lt;string&gt;</code>	Flag allowing wrapping.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.full_width</code>	<code>&lt;string&gt;</code>	Flag representing width of the layout line.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.item_spacing</code>	<code>&lt;string&gt;</code>	Item spacing between widgets.	Optional
<code>ReportTemplateSpec.img</code>	<code>&lt;object&gt;</code>	Images associated with the template.	Optional
<code>ReportTemplateSpec.img.thumbnail</code>	<code>&lt;object&gt;</code>	A thumbnail-size image for the report template.	Optional
<code>ReportTemplateSpec.img.thumbnail.src</code>	<code>&lt;string&gt;</code>	Relative URL of an image.	
<code>ReportTemplateSpec.img.full</code>	<code>&lt;object&gt;</code>	A full-size image for the report template.	Optional
<code>ReportTemplateSpec.img.full.src</code>	<code>&lt;string&gt;</code>	Relative URL of an image.	

## Reporting: List categories

Get a list of categories that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/categories
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "idx",
    "name": "index"
  },
  {
    "id": "key",
    "name": "key"
  }
]
```

Property Name	Type	Description	Notes
<i>Categories</i>	<array of <object>>	List of categories.	
<i>Categories</i> [Category]	<object>	Object representing a category.	Optional
<i>Categories</i> [Category].id	<string>	ID of a category. To be used in the API.	
<i>Categories</i> [Category].name	<string>	Human-readable name of a category.	

## Reporting: Get user attributes

Get user-specific template attributes.

```
GET https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets/{widget_id}/user_attributes
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
}
```

Example:

```
[]
```

Property Name	Type	Description	Notes
<i>TMWidgetAttributes</i>	<object>	Set of widget attributes.	
<i>TMWidgetAttributes</i> .pan_zoomable	<string>	Flag making the graph interactive.	Optional

<i>TMWidgetAttributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgetAttributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgetAttributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidgetAttributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidgetAttributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgetAttributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgetAttributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgetAttributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidgetAttributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidgetAttributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgetAttributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidgetAttributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgetAttributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgetAttributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgetAttributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidgetAttributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidgetAttributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidgetAttributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgetAttributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgetAttributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgetAttributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional

## Reporting: List statistics

Get a list of statistics that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/statistics
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "tot",
    "name": "total"
  },
  {
    "id": "avg",
    "name": "average"
  },
  {
    "id": "pek",
    "name": "peak"
  },
  {
    "id": "min",
    "name": "min"
  }
]
```

Property Name	Type	Description	Notes
<i>Statistics</i>	<array of <object>>	List of statistics.	
<i>Statistics</i> [ <i>Statistic</i> ]	<object>	Object representing a statistic.	Optional
<i>Statistics</i> [ <i>Statistic</i> ].id	<string>	ID of a statistic. To be used in the API.	
<i>Statistics</i> [ <i>Statistic</i> ].name	<string>	Human-readable name of a statistic.	

## Reporting: Set section layout

Set the layout of widgets in a grid in the template section.

PUT [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/sections/{section\\_id}/layout](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/layout)

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
```

Example:

```
[ ]
```

Property Name	Type	Description	Notes
<i>TMFlowLines</i>	<array of <object>>	Object representing visual layout of widgets in a section.	
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ]	<object>	One horizontal line of widgets.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items	<array of <object>>	List of line items.	Optional

<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items [ <i>TMFlowItem</i> ]	<object>	Object representing one layout item.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items [ <i>TMFlowItem</i> ].id	<number>	Widget ID.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes	<object>	List of line attributes.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes.item_spacing	<string>	Item spacing between widgets.	Optional

## Response Body

On success, the server does not provide any body in the responses.

## Reporting: List widgets

Get the widgets located in the template section.

```
GET https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "config": {
      "datasource": string,
      "visualization": string,
      "widget_type": string
    },
    "widget_id": number,
    "criteria": {
      "ports": [
        {
          "port": number,
          "protocol": number,
          "name": string
        }
      ],
      "dscp_app_ports": [
        {
          "port": {
            "port": number,
            "protocol": number,
            "name": string
          },
          "app": {
            "code": string,
            "name": string,
            "tunneled": string
          },
          "dscp": {
            "name": string,
            "code_point": number
          }
        }
      ],
      "services": [
        {
          "name": string,
          "service_id": number
        }
      ],
      "port_groups": [
        {
          "name": string,
          "group_id": number
        }
      ],
      "comparison_time_frame": {
        "data_resolution": string,
```



```
"refresh_interval": string,
"type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
```

```

    }
  ],
  "time_frame": {
    "data_resolution": string,
    "refresh_interval": string,
    "type": string
  },
  "service": {
    "name": string,
    "service_id": number
  },
  "severity": number,
  "role": string,
  "event_policies": [
    number
  ],
  "service_locations": [
    {
      "name": string,
      "location_id": string
    }
  ],
  "case_insensitive": string,
  "service_location": {
    "name": string,
    "location_id": string
  },
  "include_backend_segments": string,
  "host_group_type": string,
  "host_pair_app_ports": [
    {
      "port": {
        "port": number,
        "protocol": number,
        "name": string
      },
      "app": {
        "code": string,
        "name": string,
        "tunneled": string
      },
      "server": {
        "mac": string,
        "ipaddr": string,
        "name": string
      },
      "client": {
        "mac": string,
        "ipaddr": string,
        "name": string
      }
    }
  ],
  "users": [
    {
      "name": string
    }
  ],
  "sort_desc": string,
  "sort_column": number,
  "host_group_pair_ports": [
    {
      "port": {
        "port": number,
        "protocol": number,
        "name": string
      },
      "server": {
        "name": string,
        "group_id": number
      },
      "client": {
        "name": string,
        "group_id": number
      }
    }
  ],
  "network_segments": [
    {
      "src": {
        "ipaddr": string,
        "name": string,
        "ifindex": number
      },
      "dst": {
        "ipaddr": string,
        "name": string
      }
    }
  ]
}

```

```
    "name": string,
    "ifindex": number
  }
},
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
```

```

"pan_zoomable": string,
"line_scale": string,
"format_bytes": string,
"show_images": string,
"open_nodes": [
  string
],
"line_style": string,
"layout": string,
"width": number,
"height": number,
"percent_of_total": string,
"edge_thickness": string,
"display_host_group_type": string,
"extend_to_zero": string,
"collapsible": string,
"high_threshold": string,
"n_items": number,
"colspan": number,
"low_threshold": string,
"moveable_nodes": string,
"orientation": string,
"modal_links": number
},
"timestamp": string
}
]

```

Example:

```

[
{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}
]

```

Property Name	Type	Description	Notes
<i>TMWidgets</i>	<array of <object>>	List of TMWidget objects.	
<i>TMWidgets</i> [TMWidget]	<object>	One TMWidget object.	Optional
<i>TMWidgets</i> [TMWidget].config	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidgets</i> [TMWidget].config.datasource	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidgets</i> [TMWidget].config.visualization	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE

<i>TMWidgets</i> [TMWidget].config.widget_type	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidgets</i> [TMWidget].widget_id	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidgets</i> [TMWidget].criteria	<object>	Query criteria for the widget.	
<i>TMWidgets</i> [TMWidget].criteria.ports	<array of <object>>	Watched ports.	Optional
<i>TMWidgets</i> [TMWidget].criteria.ports [CProtoPort]	<object>	One CProtoPort object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.ports [CProtoPort].port	<number>	Port specification.	Optional
<i>TMWidgets</i> [TMWidget].criteria.ports [CProtoPort].protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [TMWidget].criteria.ports [CProtoPort].name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort]	<object>	One CDSCPAppPort object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port	<object>	Port specification.	
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app	<object>	Application specification.	
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp	<object>	DSCP specification.	
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [TMWidget].criteria.services	<array of <object>>	Watched services.	Optional
<i>TMWidgets</i> [TMWidget].criteria.services [CService]	<object>	One CService object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.services [CService].name	<string>	Service name.	
<i>TMWidgets</i> [TMWidget].criteria.services [CService].service_id	<number>	Service ID.	Optional
<i>TMWidgets</i> [TMWidget].criteria.port_groups	<array of <object>>	Watched port groups.	Optional
<i>TMWidgets</i> [TMWidget].criteria.port_groups [CPortGroup]	<object>	One CPortGroup object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.port_groups [CPortGroup].name	<string>	Name of the port group.	Optional
<i>TMWidgets</i> [TMWidget].criteria.port_groups [CPortGroup].group_id	<number>	ID of the port group.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs	<array of <object>>	Watched group pairs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair]	<object>	One CHostGroupPair object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server	<object>	Server host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client	<object>	Client host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.wan_group	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.traffic_expression	<string>	Traffic expression.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.split_direction	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_successes	<string>	Include successful requests in active directory report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.columns	<array of <number>>	List of column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.columns[item]	<number>	Column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server	<object>	Server specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers [CApplicationServer].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice]	<object>	One CDevice object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice].ipaddr	<string>	Device IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice].name	<string>	Device name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort]	<object>	One CApplicationPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server	<object>	Server host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].client	<object>	Client host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].client.mac	<string>	Host MAC address.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[ <i>CHostPairPort</i> ].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[ <i>CHostPairPort</i> ].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ]	<object>	One <i>CDSCPInterface</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].interface	<object>	Interface specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].interface.name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].dscp	<object>	DSCP specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].dscp.name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscp_interfaces[ <i>CDSCPInterface</i> ].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.time_frame	<object>	Widget time frame specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service	<object>	Watched service.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service.name	<string>	Service name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service.service_id	<number>	Service ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.severity	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.role	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.event_policies	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.event_policies[item]	<number>	Event policy ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_locations	<array of <object>>	Watched service locations.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_locations[ <i>CServiceLocation</i> ]	<object>	One <i>CServiceLocation</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_locations[ <i>CServiceLocation</i> ].name	<string>	Service location name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_locations[ <i>CServiceLocation</i> ].location_id	<string>	Service location ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.case_insensitive	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_location	<object>	Watched service location.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_location.name	<string>	Service location name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.service_location.location_id	<string>	Service location ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional



<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_type	<string>	Host group type used.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].server	<object>	Server host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].client	<object>	Client host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users	<array of <object>>	Watched users.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users [CUser]	<object>	One CUser object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users [CUser].name	<string>	Active Directory user name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.sort_desc	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.sort_column	<number>	Sorting column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].port.port	<number>	Port specification.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server	<object>	Server host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client	<object>	Client host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments	<array of <object>>	Watched network segments.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ]	<object>	One <i>CNetworkSegment</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src	<object>	Segment source.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst	<object>	Segment destination.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts	<array of <object>>	Watched hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ]	<object>	One <i>CHost</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs	<array of <object>>	Watched host pairs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ]	<object>	One <i>CHostPair</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server	<object>	Specification of the server host.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.name	<string>	Host name.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client	<object>	Specification of the client host.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ]	<object>	Object representing Protocol information.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ].id	<number>	ID of the Protocol.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ].name	<string>	Name of the Protocol.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces	<array of <object>>	Watched interfaces.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ]	<object>	One C <i>Interface</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups	<array of <object>>	Watched host groups.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ]	<object>	One C <i>HostGroup</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ].name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ].group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ]	<object>	One C <i>CDSCP</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ].name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ].code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications	<array of <object>>	Watched applications.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications[C <i>Application</i> ]	<object>	One C <i>Application</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications[C <i>Application</i> ].code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications[C <i>Application</i> ].name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications[C <i>Application</i> ].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].title	<string>	Widget title.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes	<object>	Widget common attributes.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.width	<number>	Widget width.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.height	<number>	Widget height.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes	<object>	User-specific attributes.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.width	<number>	Widget width.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.height	<number>	Widget height.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].timestamp	<string>	Widget time stamp specification.	Optional

## Reporting: Delete report

Delete a report.

```
DELETE https://{device}/api/profiler/1.2/reporting/reports/{report_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Reporting: List realms

Get a list of realms.

```
GET https://{device}/api/profiler/1.2/reporting/realms
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "traffic_summary",
    "name": "traffic summary"
  },
  {
    "id": "traffic_flow_list",
    "name": "traffic flow list"
  },
  {
    "id": "traffic_overall_time_series",
    "name": "traffic overall time series"
  }
]
```

Property Name	Type	Description	Notes
<i>Realms</i>	<array of <object>>	List of type of report queries (realms) that could be requested and run.	
<i>Realms</i> [ <i>Realm</i> ]	<object>	Object representing a realm.	Optional
<i>Realms</i> [ <i>Realm</i> ].id	<string>	ID of a realm. To be used in the API.	

Realms[Realm].name	<string>	Human-readable name of a realm.	
--------------------	----------	---------------------------------	--

---

## Reporting: Delete widget

Delete one widget from the template section.

```
DELETE https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets/{widget_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

---

## Reporting: Update widget

Update one widget in the template section.

```
PUT https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets/{widget_id}
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "dscp_app_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        },
        "dscp": {
          "name": string,
          "code_point": number
        }
      }
    ],
    "services": [
      {
        "name": string,
        "service_id": number
      }
    ],
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "comparison_time_frame": {
      "data_resolution": string
    }
  }
}
```

```
data_resolution": string,
"refresh_interval": string,
"type": string
},
"host_group_pairs": [
{
"server": {
"name": string,
"group_id": number
},
"client": {
"name": string,
"group_id": number
}
}
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
number
],
"application_servers": [
{
"app": {
"code": string,
"name": string,
"tunneled": string
},
"server": {
"mac": string,
"ipaddr": string,
"name": string
}
}
],
"devices": [
{
"ipaddr": string,
"name": string
}
],
"application_ports": [
{
"port": {
"port": number,
"protocol": number,
"name": string
},
"app": {
"code": string,
"name": string,
"tunneled": string
}
}
],
"include_failures": string,
"host_pair_ports": [
{
"port": {
"port": number,
"protocol": number,
"name": string
},
"server": {
"mac": string,
"ipaddr": string,
"name": string
},
"client": {
"mac": string,
"ipaddr": string,
"name": string
}
}
],
"dscp_interfaces": [
{
"interface": {
"ipaddr": string,
"name": string,
"ifindex": number
},
"dscp": {
"name": string,
"code point": number
}
```

```
    }
  }
},
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
```



```
    "name": string,
    "ifindex": number
  }
},
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
```

```

"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}

```

Property Name	Type	Description	Notes
<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE

<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort]</i>	<object>	One CDSCAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMWidget.criteria.services[CService].name</i>	<string>	Service name.	
<i>TMWidget.criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month

<i>TMWidget.criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidget.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMWidget.criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidget.criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMWidget.criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidget.criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMWidget.criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>TMWidget.criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMWidget.criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>TMWidget.criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>TMWidget.criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>TMWidget.criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.application_ports [CApplicationPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional

<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app	<object>	Application specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame</i> . data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> . refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> .type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional
<i>TMWidget.criteria.service</i> .name	<string>	Service name.	

<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation].name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_locations [CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users[CUser]</i>	<object>	One CUser object.	Optional
<i>TMWidget.criteria.users[CUser].name</i>	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional

<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client	<object>	Specification of the client host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.name	<string>	Host name.	Optional



<i>TMWidget.criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMWidget.criteria.protocols[CProtocol]</i>	<object>	Object representing Protocol information.	Optional
<i>TMWidget.criteria.protocols[CProtocol].id</i>	<number>	ID of the Protocol.	Optional
<i>TMWidget.criteria.protocols[CProtocol].name</i>	<string>	Name of the Protocol.	Optional
<i>TMWidget.criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMWidget.criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget.criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget.criteria.interfaces[CInterface]</i>	<object>	One CInterface object.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.interfaces[CInterface].name</i>	<string>	Interface name.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ifindex</i>	<number>	Interface index.	Optional
<i>TMWidget.criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup]</i>	<object>	One CHostGroup object.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidget.criteria.dscps[CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>TMWidget.criteria.dscps[CDSCP].name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscps[CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMWidget.criteria.applications [CAApplication]</i>	<object>	One CAApplication object.	Optional
<i>TMWidget.criteria.applications [CAApplication].code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.applications [CAApplication].name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.applications [CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.title</i>	<string>	Widget title.	
<i>TMWidget.attributes</i>	<object>	Widget common attributes.	Optional
<i>TMWidget.attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional



<i>TMWidget.attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes [item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidget.user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidget.user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional

## Response Body

On success, the server does not provide any body in the responses.

## Reporting: List centricities

Get a list of centricities that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/centricities
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "hos",
    "name": "host"
  },
  {
    "id": "int",
    "name": "interface"
  }
]
```

Property Name	Type	Description	Notes
Centricities	<array of <object>>	List of centricities.	
Centricities[Centricity]	<object>	Object representing a centricity.	Optional
Centricities[Centricity].id	<string>	ID of a centricity. To be used in the API.	
Centricities[Centricity].name	<string>	Human-readable name of a centricity.	

## Reporting: Import templates

Import reporting templates.

POST <https://{device}/api/profiler/1.2/reporting/templates/import>

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "traffic_expression": string,
    "id": number,
    "scheduled": string,
    "sharing": {
      "users": [
        number
      ]
    },
    "layout": [
      {
        "flow_items": [
          {
            "id": number
          }
        ],
        "attributes": {
          "wrappable": string,
          "full_width": string,
          "item_spacing": string
        }
      }
    ],
    "description": string,
    "user_id": number,
    "shared": string,
    "live": string,
    "last_added_section_id": number,
    "name": string,
    "last_added_widget_id": number,
    "version": string,
    "disabled": string,
    "timestamp": string,
    "sections": [
      {
```

```
"widgets": [
  {
    "config": {
      "datasource": string,
      "visualization": string,
      "widget_type": string
    },
    "widget_id": number,
    "criteria": {
      "ports": [
        {
          "port": number,
          "protocol": number,
          "name": string
        }
      ],
      "dscp_app_ports": [
        {
          "port": {
            "port": number,
            "protocol": number,
            "name": string
          },
          "app": {
            "code": string,
            "name": string,
            "tunneled": string
          },
          "dscp": {
            "name": string,
            "code_point": number
          }
        }
      ],
      "services": [
        {
          "name": string,
          "service_id": number
        }
      ],
      "port_groups": [
        {
          "name": string,
          "group_id": number
        }
      ],
      "comparison_time_frame": {
        "data_resolution": string,
        "refresh_interval": string,
        "type": string
      },
      "host_group_pairs": [
        {
          "server": {
            "name": string,
            "group_id": number
          },
          "client": {
            "name": string,
            "group_id": number
          }
        }
      ],
      "wan_group": string,
      "traffic_expression": string,
      "split_direction": string,
      "include_successes": string,
      "include_non_optimized_sites": string,
      "columns": [
        number
      ],
      "application_servers": [
        {
          "app": {
            "code": string,
            "name": string,
            "tunneled": string
          },
          "server": {
            "mac": string,
            "ipaddr": string,
            "name": string
          }
        }
      ],
      "devices": [
        {
```

```
"ipaddr": string,
"name": string
},
],
"application_ports": [
{
"port": {
"port": number,
"protocol": number,
"name": string
},
"app": {
"code": string,
"name": string,
"tunneled": string
}
}
],
"include_failures": string,
"host_pair_ports": [
{
"port": {
"port": number,
"protocol": number,
"name": string
},
"server": {
"mac": string,
"ipaddr": string,
"name": string
},
"client": {
"mac": string,
"ipaddr": string,
"name": string
}
}
],
"dscp_interfaces": [
{
"interface": {
"ipaddr": string,
"name": string,
"ifindex": number
},
"dscp": {
"name": string,
"code_point": number
}
}
],
"time_frame": {
"data_resolution": string,
"refresh_interval": string,
"type": string
},
"service": {
"name": string,
"service_id": number
},
"severity": number,
"role": string,
"event_policies": [
number
],
"service_locations": [
{
"name": string,
"location_id": string
}
],
"case_insensitive": string,
"service_location": {
"name": string,
"location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
{
"port": {
"port": number,
"protocol": number,
"name": string
},
"app": {
"code": string,
```

```
    "name": string,
    "tunneled": string
  },
  "server": {
    "mac": string,
    "ipaddr": string,
    "name": string
  },
  "client": {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
},
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
```

```
}
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
```

```

    item_spacing : string
  }
}
],
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
]

```

Example:

```

[
  {
    "layout": [
      {
        "flow_items": [
          {
            "id": 1
          }
        ]
      }
    ],
    "name": "VOIP - Call Quality and Usage",
    "user_id": 1,
    "timestamp": "1383141976.674345",
    "live": true,
    "last_added_widget_id": 6,
    "traffic_expression": "app VoIP-RTP",
    "version": "1.1",
    "shared": "Private",
    "sections": [
      {
        "widgets": [
          {
            "title": "VoIP-RTP: Applications",
            "timestamp": "1383141976.674383",
            "criteria": {
              "sort_column": 33,
              "traffic_expression": "",
              "sort_desc": true,
              "centricity": "host",
              "limit": 100,
              "columns": [
                17,
                33,
                34,
                757,
                766,
                781,
                803
              ],
              "time_frame": {
                "data_resolution": "15mins",
                "type": "last_hour",
                "refresh_interval": "15mins"
              }
            },
            "attributes": {
              "format_bytes": "UI_PREF",
              "colspan": 2,
              "n_items": 20
            },
            "config": {
              "widget_type": "APPS",
              "visualization": "TABLE",
              "datasource": "TRAFFIC"
            },
            "widget_id": 1
          },
          {
            "title": "VoIP-RTP: Traffic Quality",
            "timestamp": "1383141976.674428",
            "criteria": {
              "traffic_expression": "",
              "sort_desc": true,
              "centricity": "host",
              "columns": [
                803
              ],
              "time_frame": {
                "data_resolution": "min"
              }
            }
          }
        ]
      }
    ]
  }
]

```

```
    "data_resolution": "min",
    "type": "last_hour",
    "refresh_interval": "min"
  }
},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 1,
  "extend_to_zero": false,
  "line_scale": "LINEAR",
  "line_style": "STACKED"
},
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 2
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674459",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      781
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      33
    ],
    "time frame": {
```



```

    "data_resolution": "15mins",
    "type": "last_day",
    "refresh_interval": "15mins"
  }
},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 2,
  "extend_to_zero": false,
  "line_style": "STACKED"
},
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "host_group_type": "ByLocation",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "show_images": true,
    "layout": "HORIZONTAL_TREE",
    "colspan": 2,
    "moveable_nodes": true,
    "height": 400,
    "edge_thickness": true,
    "pan_zoomable": true,
    "n_items": 10
  },
  "config": {
    "widget_type": "HOST_GROUP_PAIRS",
    "visualization": "CONN_GRAPH",
    "datasource": "TRAFFIC"
  },
  "widget_id": 6
}
],
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      },
      {
        "id": 3
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 4
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 5
      }
    ]
  },
  {
    "flow_items": [

```

```

    {
      "id": 6
    }
  ],
  "section_id": 1
}
],
"id": 5217,
"description": ""
}
]

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpecs</i>	<array of <object>>	List of ReportTemplateSpec objects.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec]	<object>	One ReportTemplateSpecs object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].traffic_expression	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].id	<number>	ID of the report template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].scheduled	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing	<object>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing.users	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing.users[item]	<number>	User ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine]	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items[TMFlowItem]	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items[TMFlowItem].id	<number>	Widget ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes	<object>	List of line attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.item_spacing	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].description	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].user_id	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].shared	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].live	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].last_added_section_id	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].name	<string>	Human-readable name of the template.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].last_added_widget_id	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].version	<string>	Version of the specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].disabled	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].timestamp	<string>	Report time stamp (unix time).	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection]	<object>	One TMSection object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget]	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.datasources	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.visualization	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.widget_type	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS_HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].widget_id	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria	<object>	Query criteria for the widget.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort]	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort]	<object>	One CDSCPAppPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.name	<string>	Protocol + port combination name.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp	<object>	DSCP specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService]	<object>	One CService object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService].name	<string>	Service name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService].service_id	<number>	Service ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup]	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].name	<string>	Name of the port group.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].group_id	<number>	ID of the port group.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair]	<object>	One CHostGroupPair object.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server	<object>	Server host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client	<object>	Client host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.wan_group	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.traffic_expression	<string>	Traffic expression.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.split_direction	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_successes	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.columns	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.columns[item]	<number>	Column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.tunneled	<string>	Flag; is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server	<object>	Server specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.mac	<string>	Host MAC address.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice]	<object>	One CDevice object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].ipaddr	<string>	Device IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].name	<string>	Device name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort]	<object>	One CAApplicationPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CAApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.port	<number>	Port specification.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server	<object>	Server host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client	<object>	Client host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface	<object>	Interface specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame	<object>	Widget time frame specification.	Optional



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service	<object>	Watched service.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service.name	<string>	Service name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service.service_id	<number>	Service ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.severity	<number>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.role	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.event_policies	<array of <number>>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.event_policies[item]	<number>	Event policy ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation].name	<string>	Service location name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.case_insensitive	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location	<object>	Watched service location.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location.name	<string>	Service location name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location.location_id	<string>	Service location ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_type	<string>	Host group type used.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port	<object>	Port specification.	



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server	<object>	Server host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client	<object>	Client host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users[CUser]	<object>	One CUser object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users[CUser].name	<string>	Active Directory user name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.sort_desc	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.sort_column	<number>	Sorting column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments	<array of <object>>	Watched network segments.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src	<object>	Segment source.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst	<object>	Segment destination.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.name	<string>	Interface name.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts	<array of <object>>	Watched hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts[CHost]	<object>	One CHost object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts[CHost].mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts[CHost].ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts[CHost].name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs	<array of <object>>	Watched host pairs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair]	<object>	One CHostPair object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server	<object>	Specification of the server host.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client	<object>	Specification of the client host.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol]	<object>	Object representing Protocol information.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol].id	<number>	ID of the Protocol.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol].name	<string>	Name of the Protocol.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces	<array of <object>>	Watched interfaces.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface]	<object>	One CInterface object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups	<array of <object>>	Watched host groups.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup]	<object>	One CHostGroup object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup].name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup].group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP]	<object>	One CDSCP object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications	<array of <object>>	Watched applications.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CApplication]	<object>	One CApplication object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CApplication].code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CApplication].name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CApplication].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].title	<string>	Widget title.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes	<object>	Widget common attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.show_images	<string>	Flag showing images in a connection graph.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.width	<number>	Widget width.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.height	<number>	Widget height.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes	<object>	User-specific attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.width	<number>	Widget width.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.height	<number>	Widget height.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].timestamp	<string>	Widget time stamp specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].section_id	<number>	Section ID.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout	<array of <object>>	Internal section layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine]	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem]	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id	<number>	Widget ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes	<object>	List of line attributes.	Optional



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.item_spacing	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img	<object>	Images associated with the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.thumbnail	<object>	A thumbnail-size image for the report template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.thumbnail.src	<string>	Relative URL of an image.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.full	<object>	A full-size image for the report template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.full.src	<string>	Relative URL of an image.	

## Response Body

On success, the server does not provide any body in the responses.

## Reporting: List units

Get a list of units that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/units
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
[
  {
    "id": string,
    "name": string
  }
]

Example:
[
  {
    "id": "byt",
    "name": "bytes"
  },
  {
    "id": "pkt",
    "name": "packet"
  },
  {
    "id": "con",
    "name": "connections"
  }
]
```

Property Name	Type	Description	Notes
<i>Units</i>	<array of <object>>	List of units.	
<i>Units</i> [Unit]	<object>	Object representing a unit.	Optional
<i>Units</i> [Unit].id	<string>	ID of a unit. To be used in the API.	
<i>Units</i> [Unit].name	<string>	Human-readable name of a unit.	

## Reporting: Create template

Create a new reporting template.

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
{
  "traffic_expression": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "layout": [
    {
      "flow_items": [
        {
          "id": number
        }
      ],
      "attributes": {
        "wrappable": string,
        "full_width": string,
        "item_spacing": string
      }
    }
  ],
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "last_added_section_id": number,
  "name": string,
  "last_added_widget_id": number,
  "version": string,
  "disabled": string,
  "timestamp": string,
  "sections": [
    {
      "widgets": [
        {
          "config": {
            "datasource": string,
            "visualization": string,
            "widget_type": string
          },
          "widget_id": number,
          "criteria": {
            "ports": [
              {
                "port": number,
                "protocol": number,
                "name": string
              }
            ]
          },
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,
                "name": string
              },
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              },
              "dscp": {
                "name": string,
                "code_point": number
              }
            }
          ],
          "services": [
            {
              "name": string,
              "service_id": number
            }
          ]
        }
      ]
    }
  ]
}
```



```
    service_id: number
  }
],
"port_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"comparison_time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
]
```

```
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
]
```

```
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
```

```

    "extend_to_zero": string,
    "collapsible": string,
    "high_threshold": string,
    "n_items": number,
    "colspan": number,
    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "user_attributes": {
    "pan_zoomable": string,
    "line_scale": string,
    "format_bytes": string,
    "show_images": string,
    "open_nodes": [
      string
    ],
    "line_style": string,
    "layout": string,
    "width": number,
    "height": number,
    "percent_of_total": string,
    "edge_thickness": string,
    "display_host_group_type": string,
    "extend_to_zero": string,
    "collapsible": string,
    "high_threshold": string,
    "n_items": number,
    "colspan": number,
    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
}
],
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
}

```

Example:

```

{
  "layout": [
    {
      "flow_items": [
        {
          "id": 1
        }
      ]
    }
  ],
  "name": "VOIP - Call Quality and Usage",
  "user_id": 1,
  "timestamp": "1383141976.674345",
  "live": true,
  "last_added_widget_id": 6,
  "traffic_expression": "app VoIP-RTP",
  "version": "1.1",
  "shared": "Private",
  "sections": [
    {
      "widgets": [

```

```
{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674428",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      803
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_scale": "LINEAR",
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 2
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674459",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      781
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
}
```

```
,
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      33
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_day",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "host_group_type": "ByLocation",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "show_images": true,
    "layout": "HORIZONTAL_TREE",
    "colspan": 2,
    "moveable_nodes": true
  }
}
```

```

    "movable_nodes": true,
    "height": 400,
    "edge_thickness": true,
    "pan_zoomable": true,
    "n_items": 10
  },
  "config": {
    "widget_type": "HOST_GROUP_PAIRS",
    "visualization": "CONN_GRAPH",
    "datasource": "TRAFFIC"
  },
  "widget_id": 6
},
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      },
      {
        "id": 3
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 4
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 5
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 6
      }
    ]
  }
],
"section_id": 1
}
],
"id": 5217,
"description": ""
}

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpec</i>	<object>	Reporting template specification object.	
<i>ReportTemplateSpec.traffic_expression</i>	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpec.id</i>	<number>	ID of the report template.	Optional
<i>ReportTemplateSpec.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpec.sharing</i>	<object>	List of the users the template is shared with (see <i>ReportTemplateSharing</i> ).	Optional
<i>ReportTemplateSpec.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpec.sharing.users[item]</i>	<number>	User ID.	Optional
<i>ReportTemplateSpec.layout</i>	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional

<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.description</i>	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpec.user_id</i>	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpec.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpec.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpec.last_added_section_id</i>	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpec.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplateSpec.last_added_widget_id</i>	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpec.version</i>	<string>	Version of the specification.	Optional
<i>ReportTemplateSpec.disabled</i>	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpec.timestamp</i>	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpec.sections</i>	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpec.sections[TMSection]</i>	<object>	One TMSection object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget]</i>	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.datasources</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort]</i>	<object>	One CDSCAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService]</i>	<object>	One CService object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', 'WAN/Optimized', 'WAN/Non-optimized'.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns [item]</i>	<number>	Column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched devices.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice]</i>	<i>&lt;object&gt;</i>	One CDevice object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].ipaddr</i>	<i>&lt;string&gt;</i>	Device IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].name</i>	<i>&lt;string&gt;</i>	Device name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort]</i>	<i>&lt;object&gt;</i>	One CAApplicationPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port</i>	<i>&lt;object&gt;</i>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.port</i>	<i>&lt;number&gt;</i>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.protocol</i>	<i>&lt;number&gt;</i>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.name</i>	<i>&lt;string&gt;</i>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app</i>	<i>&lt;object&gt;</i>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.code</i>	<i>&lt;string&gt;</i>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.name</i>	<i>&lt;string&gt;</i>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.tunneled</i>	<i>&lt;string&gt;</i>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_failures</i>	<i>&lt;string&gt;</i>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort]</i>	<i>&lt;object&gt;</i>	One CHostPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port</i>	<i>&lt;object&gt;</i>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.port</i>	<i>&lt;number&gt;</i>	Port specification.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface]</i>	<object>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface</i>	<object>	Interface specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<i>&lt;string&gt;</i>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.type</i>	<i>&lt;string&gt;</i>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service</i>	<i>&lt;object&gt;</i>	Watched service.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.name</i>	<i>&lt;string&gt;</i>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.service_id</i>	<i>&lt;number&gt;</i>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.severity</i>	<i>&lt;number&gt;</i>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.role</i>	<i>&lt;string&gt;</i>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies</i>	<i>&lt;array of &lt;number&gt;&gt;</i>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies[item]</i>	<i>&lt;number&gt;</i>	Event policy ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched service locations.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<i>&lt;object&gt;</i>	One CServiceLocation object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<i>&lt;string&gt;</i>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<i>&lt;string&gt;</i>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.case_insensitive</i>	<i>&lt;string&gt;</i>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location</i>	<i>&lt;object&gt;</i>	Watched service location.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.name</i>	<i>&lt;string&gt;</i>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.location_id</i>	<i>&lt;string&gt;</i>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_backend_segments</i>	<i>&lt;string&gt;</i>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_type</i>	<i>&lt;string&gt;</i>	Host group type used.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<i>&lt;object&gt;</i>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<i>&lt;object&gt;</i>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<i>&lt;number&gt;</i>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<i>&lt;number&gt;</i>	Protocol specification.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser]</i>	<object>	One CUser object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser].name</i>	<string>	Active Directory user name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]</i>	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port</i>	<object>	Port specification.	



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port</i>	<i>&lt;number&gt;</i>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol</i>	<i>&lt;number&gt;</i>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name</i>	<i>&lt;string&gt;</i>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server</i>	<i>&lt;object&gt;</i>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client</i>	<i>&lt;object&gt;</i>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched network segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<i>&lt;object&gt;</i>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<i>&lt;object&gt;</i>	Segment source.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<i>&lt;object&gt;</i>	Segment destination.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]</i>	<i>&lt;object&gt;</i>	One CHost object.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair]</i>	<i>&lt;object&gt;</i>	One CHostPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server</i>	<i>&lt;object&gt;</i>	Specification of the server host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client</i>	<i>&lt;object&gt;</i>	Specification of the client host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched protocols.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol]</i>	<i>&lt;object&gt;</i>	Object representing Protocol information.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].id</i>	<i>&lt;number&gt;</i>	ID of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].name</i>	<i>&lt;string&gt;</i>	Name of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.centricity</i>	<i>&lt;string&gt;</i>	Centricity used to run the report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.limit</i>	<i>&lt;number&gt;</i>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface]</i>	<i>&lt;object&gt;</i>	One CInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup]</i>	<i>&lt;object&gt;</i>	One CHostGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups[HostGroup].group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched DSCPs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP]</i>	<i>&lt;object&gt;</i>	One CDSCP object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].name</i>	<i>&lt;string&gt;</i>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].code_point</i>	<i>&lt;number&gt;</i>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched applications.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication]</i>	<i>&lt;object&gt;</i>	One CApplication object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].code</i>	<i>&lt;string&gt;</i>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].name</i>	<i>&lt;string&gt;</i>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].tunneled</i>	<i>&lt;string&gt;</i>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].title</i>	<i>&lt;string&gt;</i>	Widget title.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes</i>	<i>&lt;object&gt;</i>	Widget common attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].section_id</i>	<i>&lt;number&gt;</i>	Section ID.	
<i>ReportTemplateSpec.sections[TMSection].layout</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Internal section layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine]</i>	<i>&lt;object&gt;</i>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of line items.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem]</i>	<i>&lt;object&gt;</i>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<i>&lt;number&gt;</i>	Widget ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes</i>	<i>&lt;object&gt;</i>	List of line attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.wrappable</i>	<i>&lt;string&gt;</i>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.full_width</i>	<i>&lt;string&gt;</i>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.item_spacing</i>	<i>&lt;string&gt;</i>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.img</i>	<i>&lt;object&gt;</i>	Images associated with the template.	Optional
<i>ReportTemplateSpec.img.thumbnail</i>	<i>&lt;object&gt;</i>	A thumbnail-size image for the report template.	Optional
<i>ReportTemplateSpec.img.thumbnail.src</i>	<i>&lt;string&gt;</i>	Relative URL of an image.	
<i>ReportTemplateSpec.img.full</i>	<i>&lt;object&gt;</i>	A full-size image for the report template.	Optional
<i>ReportTemplateSpec.img.full.src</i>	<i>&lt;string&gt;</i>	Relative URL of an image.	

## Response Body

On success, the server returns a response body with the following structure:

JSON

```

{
  "traffic_expression": string,
  "schedule_type": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "name": string,
  "disabled": string,
  "next_run": number
}

```

Example:

```

{
  "user_id": 1,
  "live": true,
  "id": 1000,
  "name": "My Template"
}

```

Property Name	Type	Description	Notes
<i>ReportTemplate</i>	<object>	A template for running reports.	
<i>ReportTemplate.traffic_expression</i>	<string>	Traffic expression applied to all widgets within this template.	Optional
<i>ReportTemplate.schedule_type</i>	<string>	Type of template scheduling.	Optional; Values: Once, Hourly, Daily, Weekly, Monthly, Quarterly
<i>ReportTemplate.id</i>	<number>	ID of the template.	
<i>ReportTemplate.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplate.sharing</i>	<object>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplate.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplate.sharing.users[item]</i>	<number>	User ID.	Optional
<i>ReportTemplate.description</i>	<string>	Description of the template.	Optional
<i>ReportTemplate.user_id</i>	<number>	ID of the user who owns the template.	Optional
<i>ReportTemplate.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplate.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplate.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplate.disabled</i>	<string>	Flag indicating that data collection for the template is disabled.	Optional
<i>ReportTemplate.next_run</i>	<number>	Next run time for the template if the template is scheduled to run.	Optional

## Reporting: Get reporting end times

List end times for each data source.

```
GET https://{device}/api/profiler/1.2/reporting/timestamps
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "data_resolution": string,
    "end_time": number,
    "datasource": string
  }
]
```

Example:

```
[
  {
    "data_resolution": "day",
    "end_time": 1383105600,
    "datasource": "TRAFFIC"
  },
  {
    "data_resolution": "15mins",
    "end_time": 1383160500,
    "datasource": "SERVICE"
  },
  {
    "data_resolution": "min",
    "end_time": 1383161280,
    "datasource": "TRAFFIC"
  },
  {
    "data_resolution": "6hours",
    "end_time": 1383148800,
    "datasource": "TRAFFIC"
  },
  {
    "data_resolution": "hour",
    "end_time": 1383159600,
    "datasource": "TRAFFIC"
  },
  {
    "data_resolution": "15mins",
    "end_time": 1383160500,
    "datasource": "TRAFFIC"
  }
]
```

Property Name	Type	Description	Notes
<i>ReportTimestamps</i>	<array of <object>>	Collection of ReportTimestamp objects.	
<i>ReportTimestamps</i> [ReportTimestamp]	<object>	Object representing report timing information.	Optional
<i>ReportTimestamps</i> [ReportTimestamp].data_resolution	<string>	Report data resolution. (can be: 1min, 15min, hour, 6hour, day, week, month).	Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTimestamps</i> [ReportTimestamp].end_time	<number>	Report end time (unix time).	
<i>ReportTimestamps</i> [ReportTimestamp].datasource	<string>	Report data source type (can be: TRAFFIC, SERVICE, EVENTS, ACTIVE_DIRECTORY).	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY

## Reporting: List severities

Get a list of severities that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/severities
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "id": string,
    "name": string
  }
]

Example:
[
  {
    "id": "nav",
    "name": "not available"
  },
  {
    "id": "nml",
    "name": "normal"
  },
  {
    "id": "low",
    "name": "low"
  },
  {
    "id": "med",
    "name": "medium"
  },
  {
    "id": "hgh",
    "name": "high"
  },
  {
    "id": "all",
    "name": "all"
  }
]
```

Property Name	Type	Description	Notes
<i>Severities</i>	<array of <object>>	List of severities.	
<i>Severities</i> [Severity]	<object>	Object representing a severity.	Optional
<i>Severities</i> [Severity].id	<string>	ID of a severity. To be used in the API.	
<i>Severities</i> [Severity].name	<string>	Human-readable name of a severity.	

## Reporting: Get query data

Get data for one or many columns from this query.

```
GET https://{device}/api/profiler/1.2/reporting/reports/{report_id}/queries/{query_id}?columns={string}&offset={number}&limit={number}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>columns</i>	<string>	Comma-separated list of column ids.	Optional
<i>offset</i>	<number>	Start row.	Optional
<i>limit</i>	<number>	Number of rows to be returned.	Optional

### Response Body

On success, the server returns a response body with the following structure:

JSON

```

{
  "data": [
    [
      string
    ]
  ],
  "data_size": number,
  "totals": [
    string
  ]
}

```

Example:

```

{
  "data": [
    [
      "10.38.8.202|",
      "6878717.15556",
      "7041.06111111"
    ],
    [
      "10.38.9.152|",
      "1996165.24167",
      "2049.01388889"
    ]
  ],
  "data_size": 3744,
  "totals": [
    "",
    "20293913.8417",
    "23577.3055556"
  ]
}

```

Property Name	Type	Description	Notes
<i>DataResults</i>	<object>	Object representing a 2-dimensional array of query data and totals.	
<i>DataResults.data</i>	<array of <array of <string>>>	Two-dimensional data array.	
<i>DataResults.data[Row]</i>	<array of <string>>	One row in the list of rows.	Optional
<i>DataResults.data[Row][item]</i>	<string>	One value datum.	Optional
<i>DataResults.data_size</i>	<number>	Number of rows in the data array.	Optional
<i>DataResults.totals</i>	<array of <string>>	Object representing a row of total values (totals).	Optional
<i>DataResults.totals[item]</i>	<string>	One total datum.	Optional

## Reporting: List group bys

Get a list of reporting summarizations (group bys).

```
GET https://{device}/api/profiler/1.2/reporting/group_bys
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "hos",
    "name": "host"
  },
  {
    "id": "hop",
    "name": "host pair"
  },
  {
    "id": "gro",
    "name": "host group"
  },
  {
    "id": "gpp",
    "name": "host group pair"
  }
]
```

Property Name	Type	Description	Notes
GroupBys	<array of <object>>	List of reporting summarizations (group-bys).	
GroupBys[GroupBy]	<object>	Object representing a group by.	Optional
GroupBys[GroupBy].id	<string>	ID of a group by. To be used in the API.	
GroupBys[GroupBy].name	<string>	Human-readable name of a group by.	

## Reporting: Create widget

Create a new widget in the template section.

POST [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/sections/{section\\_id}/widgets](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets)

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ]
  },
  "dscp_app_ports": [
    {
      "port": {
        "port": number,
        "protocol": number,
        "name": string
      },
      "app": {
        "code": string,
        "name": string,
        "tunneled": string
      }
    }
  ],
  "dscp": {
```



```
    "name": string,  
    "code_point": number  
  }  
},  
],  
"services": [  
  {  
    "name": string,  
    "service_id": number  
  }  
],  
"port_groups": [  
  {  
    "name": string,  
    "group_id": number  
  }  
],  
"comparison_time_frame": {  
  "data_resolution": string,  
  "refresh_interval": string,  
  "type": string  
},  
"host_group_pairs": [  
  {  
    "server": {  
      "name": string,  
      "group_id": number  
    },  
    "client": {  
      "name": string,  
      "group_id": number  
    }  
  }  
],  
"wan_group": string,  
"traffic_expression": string,  
"split_direction": string,  
"include_successes": string,  
"include_non_optimized_sites": string,  
"columns": [  
  number  
],  
"application_servers": [  
  {  
    "app": {  
      "code": string,  
      "name": string,  
      "tunneled": string  
    },  
    "server": {  
      "mac": string,  
      "ipaddr": string,  
      "name": string  
    }  
  }  
],  
"devices": [  
  {  
    "ipaddr": string,  
    "name": string  
  }  
],  
"application_ports": [  
  {  
    "port": {  
      "port": number,  
      "protocol": number,  
      "name": string  
    },  
    "app": {  
      "code": string,  
      "name": string,  
      "tunneled": string  
    }  
  }  
],  
"include_failures": string,  
"host_pair_ports": [  
  {  
    "port": {  
      "port": number,  
      "protocol": number,  
      "name": string  
    },  
    "server": {  
      "mac": string,  
      "ipaddr": string,  

```

```
    "name": string
  },
  "client": {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
},
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },

```

```
"server": {
  "name": string,
  "group_id": number
},
"client": {
  "name": string,
  "group_id": number
}
},
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
```

```

    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
  },
  "time_frame": {
    "data_resolution": "15mins",
    "type": "last_hour",
    "refresh_interval": "15mins"
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}

```

Property Name	Type	Description	Notes
---------------	------	-------------	-------

<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMWidget.criteria.services[CService].name</i>	<string>	Service name.	
<i>TMWidget.criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional

<i>TMWidget.criteria.port_groups</i> [CPortGroup].group_id	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair]	<object>	One CHostGroupPair object.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs</i> [CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidget.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMWidget.criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidget.criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMWidget.criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidget.criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMWidget.criteria.columns</i> [item]	<number>	Column ID.	Optional
<i>TMWidget.criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].app	<object>	Application specification.	
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server	<object>	Server specification.	
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMWidget.criteria.devices</i> [CDevice]	<object>	One CDevice object.	Optional
<i>TMWidget.criteria.devices</i> [CDevice].ipaddr	<string>	Device IP address.	Optional
<i>TMWidget.criteria.devices</i> [CDevice].name	<string>	Device name.	Optional
<i>TMWidget.criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional

<i>TMWidget.criteria.application_ports</i> [CApplicationPort]	<object>	One CApplicationPort object.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port	<object>	Port specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app	<object>	Application specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame.</i> <i>data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame.</i> <i>refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month

<i>TMWidget.criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional
<i>TMWidget.criteria.service.name</i>	<string>	Service name.	
<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation].name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_locations [CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users[CUser]</i>	<object>	One CUser object.	Optional



<i>TMWidget.criteria.users</i> [CUser].name	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].client	<object>	Specification of the client host.	

<i>TMWidget</i> .criteria.host_pairs[CHostPair].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget</i> .criteria.host_pairs[CHostPair].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget</i> .criteria.host_pairs[CHostPair].client.name	<string>	Host name.	Optional
<i>TMWidget</i> .criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>TMWidget</i> .criteria.protocols[CProtocol]	<object>	Object representing Protocol information.	Optional
<i>TMWidget</i> .criteria.protocols[CProtocol].id	<number>	ID of the Protocol.	Optional
<i>TMWidget</i> .criteria.protocols[CProtocol].name	<string>	Name of the Protocol.	Optional
<i>TMWidget</i> .criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>TMWidget</i> .criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget</i> .criteria.interfaces	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget</i> .criteria.interfaces[CInterface]	<object>	One CInterface object.	Optional
<i>TMWidget</i> .criteria.interfaces[CInterface].ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget</i> .criteria.interfaces[CInterface].name	<string>	Interface name.	Optional
<i>TMWidget</i> .criteria.interfaces[CInterface].ifindex	<number>	Interface index.	Optional
<i>TMWidget</i> .criteria.host_groups	<array of <object>>	Watched host groups.	Optional
<i>TMWidget</i> .criteria.host_groups [CHostGroup]	<object>	One CHostGroup object.	Optional
<i>TMWidget</i> .criteria.host_groups [CHostGroup].name	<string>	Host group name.	Optional
<i>TMWidget</i> .criteria.host_groups [CHostGroup].group_id	<number>	Host group ID.	Optional
<i>TMWidget</i> .criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidget</i> .criteria.dscps[CDSCP]	<object>	One CDSCP object.	Optional
<i>TMWidget</i> .criteria.dscps[CDSCP].name	<string>	DSCP name.	Optional
<i>TMWidget</i> .criteria.dscps[CDSCP].code_point	<number>	DSCP code point.	Optional
<i>TMWidget</i> .criteria.applications	<array of <object>>	Watched applications.	Optional
<i>TMWidget</i> .criteria.applications [CApplication]	<object>	One CApplication object.	Optional
<i>TMWidget</i> .criteria.applications [CApplication].code	<string>	Application code.	Optional
<i>TMWidget</i> .criteria.applications [CApplication].name	<string>	Application name.	Optional
<i>TMWidget</i> .criteria.applications [CApplication].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget</i> .title	<string>	Widget title.	
<i>TMWidget</i> .attributes	<object>	Widget common attributes.	Optional
<i>TMWidget</i> .attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMWidget</i> .attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget</i> .attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget</i> .attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget</i> .attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget</i> .attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget</i> .attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget</i> .attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget</i> .attributes.width	<number>	Widget width.	Optional
<i>TMWidget</i> .attributes.height	<number>	Widget height.	Optional

<i>TMWidget.attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes [item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidget.user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidget.user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number
```

```
    "port": number,  
    "protocol": number,  
    "name": string  
  },  
],  
"dscp_app_ports": [  
  {  
    "port": {  
      "port": number,  
      "protocol": number,  
      "name": string  
    },  
    "app": {  
      "code": string,  
      "name": string,  
      "tunneled": string  
    },  
    "dscp": {  
      "name": string,  
      "code_point": number  
    }  
  }  
],  
"services": [  
  {  
    "name": string,  
    "service_id": number  
  }  
],  
"port_groups": [  
  {  
    "name": string,  
    "group_id": number  
  }  
],  
"comparison_time_frame": {  
  "data_resolution": string,  
  "refresh_interval": string,  
  "type": string  
},  
"host_group_pairs": [  
  {  
    "server": {  
      "name": string,  
      "group_id": number  
    },  
    "client": {  
      "name": string,  
      "group_id": number  
    }  
  }  
],  
"wan_group": string,  
"traffic_expression": string,  
"split_direction": string,  
"include_successes": string,  
"include_non_optimized_sites": string,  
"columns": [  
  number  
],  
"application_servers": [  
  {  
    "app": {  
      "code": string,  
      "name": string,  
      "tunneled": string  
    },  
    "server": {  
      "mac": string,  
      "ipaddr": string,  
      "name": string  
    }  
  }  
],  
"devices": [  
  {  
    "ipaddr": string,  
    "name": string  
  }  
],  
"application_ports": [  
  {  
    "port": {  
      "port": number,  
      "protocol": number,  
      "name": string  
    },  
    "app": {
```

```
    "code": string,
    "name": string,
    "tunneled": string
  }
}
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
]
```

```
    }
  },
  ],
  "users": [
    {
      "name": string
    }
  ],
  "sort_desc": string,
  "sort_column": number,
  "host_group_pair_ports": [
    {
      "port": {
        "port": number,
        "protocol": number,
        "name": string
      },
      "server": {
        "name": string,
        "group_id": number
      },
      "client": {
        "name": string,
        "group_id": number
      }
    }
  ],
  "network_segments": [
    {
      "src": {
        "ipaddr": string,
        "name": string,
        "ifindex": number
      },
      "dst": {
        "ipaddr": string,
        "name": string,
        "ifindex": number
      }
    }
  ],
  "hosts": [
    {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  ],
  "host_pairs": [
    {
      "server": {
        "mac": string,
        "ipaddr": string,
        "name": string
      },
      "client": {
        "mac": string,
        "ipaddr": string,
        "name": string
      }
    }
  ],
  "protocols": [
    {
      "id": number,
      "name": string
    }
  ],
  "centricity": string,
  "limit": number,
  "interfaces": [
    {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  ],
  "host_groups": [
    {
      "name": string,
      "group_id": number
    }
  ],
  "dscps": [
    {
      "name": string,
      "code_point": number
    }
  ]
}
```

```

    }
  ],
  "applications": [
    {
      "code": string,
      "name": string,
      "tunneled": string
    }
  ]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
  },
  "time_frame": {
    "data_resolution": "15mins",
    "type": "last_hour",
    "refresh_interval": "15mins"
  }
}

```

```

},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 2,
  "n_items": 20
},
"config": {
  "widget_type": "APPS",
  "visualization": "TABLE",
  "datasource": "TRAFFIC"
},
"widget_id": 1
}

```

Property Name	Type	Description	Notes
<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional



<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMWidget.criteria.services[CService].name</i>	<string>	Service name.	
<i>TMWidget.criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidget.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMWidget.criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidget.criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMWidget.criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidget.criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMWidget.criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>TMWidget.criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	

<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.application_servers</i> [CApplicationServer].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMWidget.criteria.devices</i> [CDevice]	<object>	One CDevice object.	Optional
<i>TMWidget.criteria.devices</i> [CDevice]. ipaddr	<string>	Device IP address.	Optional
<i>TMWidget.criteria.devices</i> [CDevice].name	<string>	Device name.	Optional
<i>TMWidget.criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort]	<object>	One CApplicationPort object.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port	<object>	Port specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app	<object>	Application specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional

<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame.</i> data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame.</i> refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional
<i>TMWidget.criteria.service.name</i>	<string>	Service name.	
<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies</i> [item]	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations</i> [CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations</i> [CServiceLocation].name	<string>	Service location name.	
<i>TMWidget.criteria.service_locations</i> [CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location.</i> location_id	<string>	Service location ID.	Optional
<i>TMWidget.criteria.</i> include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server	<object>	Server host specification.	

<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i> [CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users</i> [CUser]	<object>	One CUser object.	Optional
<i>TMWidget.criteria.users</i> [CUser].name	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional

<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].client	<object>	Specification of the client host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol]	<object>	Object representing Protocol information.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol].id	<number>	ID of the Protocol.	Optional
<i>TMWidget.criteria.protocols</i> [CProtocol].name	<string>	Name of the Protocol.	Optional
<i>TMWidget.criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMWidget.criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget.criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface]	<object>	One CInterface object.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface].ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface].name	<string>	Interface name.	Optional
<i>TMWidget.criteria.interfaces</i> [CInterface].ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup]	<object>	One CHostGroup object.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup].name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_groups</i> [CHostGroup].group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidget.criteria.dscps</i> [CDSCP]	<object>	One CDSCP object.	Optional
<i>TMWidget.criteria.dscps</i> [CDSCP].name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscps</i> [CDSCP].code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMWidget.criteria.applications</i> [CApplication]	<object>	One CApplication object.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].code	<string>	Application code.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].name	<string>	Application name.	Optional
<i>TMWidget.criteria.applications</i> [CApplication].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.title</i>	<string>	Widget title.	
<i>TMWidget.attributes</i>	<object>	Widget common attributes.	Optional
<i>TMWidget.attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional

<i>TMWidget.attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, MBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<object>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, MBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.user_attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.user_attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.user_attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.user_attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.user_attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.user_attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>TMWidget.user_attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.user_attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.user_attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.user_attributes.modal_links</i>	<number>	Flag adding modal links on a widget.	Optional

<i>TMWidget.timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional
---------------------------	-----------------------	----------------------------------	----------

## Reporting: List columns

Get a list of columns.

```
GET https://{device}/api/profiler/1.2/reporting/columns?metric={string}&statistic={string}&severity={string}&role={string}&category={string}&group_by={string}&direction={string}&area={string}&centricity={string}&unit={string}&rate={string}&realm={string}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>metric</i>	<i>&lt;string&gt;</i>	Filter the list of columns by metric.	Optional
<i>statistic</i>	<i>&lt;string&gt;</i>	Filter the list of columns by statistic.	Optional
<i>severity</i>	<i>&lt;string&gt;</i>	Filter the list of columns by severity.	Optional
<i>role</i>	<i>&lt;string&gt;</i>	Filter the list of columns by role.	Optional
<i>category</i>	<i>&lt;string&gt;</i>	Filter the list of columns by category.	Optional
<i>group_by</i>	<i>&lt;string&gt;</i>	Filter the list of columns by group by.	Optional
<i>direction</i>	<i>&lt;string&gt;</i>	Filter the list of columns by direction.	Optional
<i>area</i>	<i>&lt;string&gt;</i>	Filter the list of columns by area.	Optional
<i>centricity</i>	<i>&lt;string&gt;</i>	Filter the list of columns by centricity.	Optional
<i>unit</i>	<i>&lt;string&gt;</i>	Filter the list of columns by unit.	Optional
<i>rate</i>	<i>&lt;string&gt;</i>	Filter the list of columns by rate.	Optional
<i>realm</i>	<i>&lt;string&gt;</i>	Filter the list of columns by realm.	Optional

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```



```
[
  {
    "metric": string,
    "cli_srv": string,
    "comparison_parameter": string,
    "internal": string,
    "id": number,
    "strid": string,
    "statistic": string,
    "severity": string,
    "role": string,
    "category": string,
    "name": string,
    "comparison": string,
    "sortable": string,
    "type": string,
    "direction": string,
    "available": string,
    "context": string,
    "area": string,
    "has_others": string,
    "unit": string,
    "name_type": string,
    "rate": string
  }
]
```

Example:

```
[
  {
    "strid": "ID_TOTAL_BYTES",
    "metric": "net_bw",
    "rate": "count",
    "statistic": "total",
    "id": 30,
    "unit": "bytes",
    "category": "data",
    "severity": "none",
    "area": "none",
    "internal": false,
    "role": "none",
    "cli_srv": "none",
    "type": "int",
    "available": false,
    "direction": "none",
    "comparison": "none",
    "sortable": true,
    "name": "Total Bytes",
    "comparison_parameter": "",
    "has_others": false,
    "context": false,
    "name_type": "colname_parts"
  },
  {
    "strid": "ID_TOTAL_PKTTS",
    "metric": "net_bw",
    "rate": "count",
    "statistic": "total",
    "id": 31,
    "unit": "pkts",
    "category": "data",
    "severity": "none",
    "area": "none",
    "internal": false,
    "role": "none",
    "cli_srv": "none",
    "type": "int",
    "available": false,
    "direction": "none",
    "comparison": "none",
    "sortable": true,
    "name": "Total Packets",
    "comparison_parameter": "",
    "has_others": false,
    "context": false,
    "name_type": "colname_parts"
  }
]
```

Property Name	Type	Description	Notes
Columns	<array of <object>>	List of reporting query columns.	
Columns[Column]	<object>	A column for reporting query.	Optional



<code>Columns[Column].metric</code>	<code>&lt;string&gt;</code>	Column 'metric'. See 'reporting/metrics'.	
<code>Columns[Column].cli_srv</code>	<code>&lt;string&gt;</code>	Text flag indicating if the column is for the clients or servers.	
<code>Columns[Column].comparison_parameter</code>	<code>&lt;string&gt;</code>	Parameter for column comparison.	
<code>Columns[Column].internal</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if the column is internal to the system.	
<code>Columns[Column].id</code>	<code>&lt;number&gt;</code>	System ID for the column. Used in the API.	
<code>Columns[Column].strid</code>	<code>&lt;string&gt;</code>	String ID for the column. Not used by the API, but easier for the human user to see.	
<code>Columns[Column].statistic</code>	<code>&lt;string&gt;</code>	Column 'statistic'. See 'reporting/statistics'.	
<code>Columns[Column].severity</code>	<code>&lt;string&gt;</code>	Column 'severity'. See 'reporting/severities'.	
<code>Columns[Column].role</code>	<code>&lt;string&gt;</code>	Column 'role'. See 'reporting/roles'.	
<code>Columns[Column].category</code>	<code>&lt;string&gt;</code>	Column 'category'. See 'reporting/categories'.	
<code>Columns[Column].name</code>	<code>&lt;string&gt;</code>	Column name. Format used for column names is similar to the format used for column data.	
<code>Columns[Column].comparison</code>	<code>&lt;string&gt;</code>	Column 'comparison'. See 'reporting/comparisons'.	
<code>Columns[Column].sortable</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if this data can be sorted on this column when running the template.	
<code>Columns[Column].type</code>	<code>&lt;string&gt;</code>	Type of the column data. See 'reporting/types'.	
<code>Columns[Column].direction</code>	<code>&lt;string&gt;</code>	Column 'direction'. See 'reporting/directions'.	
<code>Columns[Column].available</code>	<code>&lt;string&gt;</code>	Boolean flag indicating that the data for the column is available without the need to re-run the template.	
<code>Columns[Column].context</code>	<code>&lt;string&gt;</code>	Internal flag used for formatting certain kinds of data.	
<code>Columns[Column].area</code>	<code>&lt;string&gt;</code>	Column 'area'. See 'reporting/area'.	
<code>Columns[Column].has_others</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if the column's 'other' row can be computed.	
<code>Columns[Column].unit</code>	<code>&lt;string&gt;</code>	Column 'unit'. See 'reporting/units'.	
<code>Columns[Column].name_type</code>	<code>&lt;string&gt;</code>	Type of the column name. See 'reporting/types'.	
<code>Columns[Column].rate</code>	<code>&lt;string&gt;</code>	Column 'rate'. See 'reporting/rates'.	

## Reporting: Get section layout

Get the layout of the template section.

GET [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/sections/{section\\_id}/layout](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/layout)

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
[
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
```

Example:  
[]

Property Name	Type	Description	Notes
<code>TMFlowLines</code>	<code>&lt;array of object&gt;</code>	Object representing visual layout of widgets in a section.	

<i>TMFlowLines</i> [ <i>TMFlowLine</i> ]	<object>	One horizontal line of widgets.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items	<array of <object>>	List of line items.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items [ <i>TMFlowItem</i> ]	<object>	Object representing one layout item.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].flow_items [ <i>TMFlowItem</i> ].id	<number>	Widget ID.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes	<object>	List of line attributes.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes. wrappable	<string>	Flag allowing wrapping.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes. full_width	<string>	Flag representing width of the layout line.	Optional
<i>TMFlowLines</i> [ <i>TMFlowLine</i> ].attributes. item_spacing	<string>	Item spacing between widgets.	Optional

## Reporting: Get report

Get information for report. Includes progress information for running reports.

GET [https://{device}/api/profiler/1.2/reporting/reports/{report\\_id}](https://{device}/api/profiler/1.2/reporting/reports/{report_id})

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
{
  "run_time": number,
  "error_text": string,
  "remaining_seconds": number,
  "saved": string,
  "id": number,
  "status": string,
  "percent": number,
  "user_id": number,
  "size": number,
  "name": string,
  "template_id": number
}
```

Example:

```
{
  "status": "completed",
  "user_id": 1,
  "name": "Host Information Report",
  "percent": 100,
  "id": 1001,
  "remaining_seconds": 0,
  "run_time": 1352494550,
  "saved": true,
  "template_id": 952,
  "error_text": "",
  "size": 140
}
```

Property Name	Type	Description	Notes
<i>ReportInfo</i>	<object>	Object representing report information.	
<i>ReportInfo</i> .run_time	<number>	Time when the report was run (Unix time).	
<i>ReportInfo</i> .error_text	<string>	A report can be completed with an error. Error message may provide more detailed info.	Optional
<i>ReportInfo</i> .remaining_seconds	<number>	Number of seconds remaining to run the report. Even if this number is 0, the report may not yet be completed, so check 'status' to make sure what the status is.	
<i>ReportInfo</i> .saved	<string>	Boolean flag indicating if the report was saved.	
<i>ReportInfo</i> .id	<number>	ID of the report. To be used in the API.	
<i>ReportInfo</i> .status	<string>	Status of the report.	Values: completed, running, waiting
<i>ReportInfo</i> .percent	<number>	Progress of the report represented by percentage of report completion.	

<i>ReportInfo</i> .user_id	<number>	ID of the user who owns the report.	
<i>ReportInfo</i> .size	<number>	Size of the report in kilobytes.	
<i>ReportInfo</i> .name	<string>	Name of the report. Could be given by a user or automatically generated by the system.	Optional
<i>ReportInfo</i> .template_id	<number>	ID of the template that the report is based on.	

## Reporting: Get template configuration

Get template configuration data.

GET [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/config](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/config)

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "traffic_expression": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "layout": [
    {
      "flow_items": [
        {
          "id": number
        }
      ],
      "attributes": {
        "wrappable": string,
        "full_width": string,
        "item_spacing": string
      }
    }
  ],
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "last_added_section_id": number,
  "name": string,
  "last_added_widget_id": number,
  "version": string,
  "disabled": string,
  "timestamp": string,
  "sections": [
    {
      "widgets": [
        {
          "config": {
            "datasource": string,
            "visualization": string,
            "widget_type": string
          },
          "widget_id": number,
          "criteria": {
            "ports": [
              {
                "port": number,
                "protocol": number,
                "name": string
              }
            ]
          },
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,

```

```
    "name": string
  },
  "app": {
    "code": string,
    "name": string,
    "tunneled": string
  },
  "dscp": {
    "name": string,
    "code_point": number
  }
},
],
"services": [
  {
    "name": string,
    "service_id": number
  }
],
"port_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"comparison_time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
```

```
"port": {
  "port": number,
  "protocol": number,
  "name": string
},
"server": {
  "mac": string,
  "ipaddr": string,
  "name": string
},
"client": {
  "mac": string,
  "ipaddr": string,
  "name": string
}
},
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number
```

```
    sort_column : number,
    "host_group_pair_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "server": {
          "name": string,
          "group_id": number
        },
        "client": {
          "name": string,
          "group_id": number
        }
      }
    ],
    "network_segments": [
      {
        "src": {
          "ipaddr": string,
          "name": string,
          "ifindex": number
        },
        "dst": {
          "ipaddr": string,
          "name": string,
          "ifindex": number
        }
      }
    ],
    "hosts": [
      {
        "mac": string,
        "ipaddr": string,
        "name": string
      }
    ],
    "host_pairs": [
      {
        "server": {
          "mac": string,
          "ipaddr": string,
          "name": string
        },
        "client": {
          "mac": string,
          "ipaddr": string,
          "name": string
        }
      }
    ],
    "protocols": [
      {
        "id": number,
        "name": string
      }
    ],
    "centricity": string,
    "limit": number,
    "interfaces": [
      {
        "ipaddr": string,
        "name": string,
        "ifindex": number
      }
    ],
    "host_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "dscps": [
      {
        "name": string,
        "code_point": number
      }
    ],
    "applications": [
      {
        "code": string,
        "name": string,
        "tunneled": string
      }
    ]
  }
}
```

```

    },
    "title": string,
    "attributes": {
      "pan_zoomable": string,
      "line_scale": string,
      "format_bytes": string,
      "show_images": string,
      "open_nodes": [
        string
      ],
      "line_style": string,
      "layout": string,
      "width": number,
      "height": number,
      "percent_of_total": string,
      "edge_thickness": string,
      "display_host_group_type": string,
      "extend_to_zero": string,
      "collapsible": string,
      "high_threshold": string,
      "n_items": number,
      "colspan": number,
      "low_threshold": string,
      "moveable_nodes": string,
      "orientation": string,
      "modal_links": number
    },
    "user_attributes": {
      "pan_zoomable": string,
      "line_scale": string,
      "format_bytes": string,
      "show_images": string,
      "open_nodes": [
        string
      ],
      "line_style": string,
      "layout": string,
      "width": number,
      "height": number,
      "percent_of_total": string,
      "edge_thickness": string,
      "display_host_group_type": string,
      "extend_to_zero": string,
      "collapsible": string,
      "high_threshold": string,
      "n_items": number,
      "colspan": number,
      "low_threshold": string,
      "moveable_nodes": string,
      "orientation": string,
      "modal_links": number
    },
    "timestamp": string
  }
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
},
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
}

```

Example:

```

{
  "layout": [
    {
      "flow_items": [
        {

```

```
    "id": 1
  }
]
},
"name": "VOIP - Call Quality and Usage",
"user_id": 1,
"timestamp": "1383141976.674345",
"live": true,
"last_added_widget_id": 6,
"traffic_expression": "app VoIP-RTP",
"version": "1.1",
"shared": "Private",
"sections": [
  {
    "widgets": [
      {
        "title": "VoIP-RTP: Applications",
        "timestamp": "1383141976.674383",
        "criteria": {
          "sort_column": 33,
          "traffic_expression": "",
          "sort_desc": true,
          "centricity": "host",
          "limit": 100,
        },
        "columns": [
          17,
          33,
          34,
          757,
          766,
          781,
          803
        ],
        "time_frame": {
          "data_resolution": "15mins",
          "type": "last_hour",
          "refresh_interval": "15mins"
        },
        "attributes": {
          "format_bytes": "UI_PREF",
          "colspan": 2,
          "n_items": 20
        },
        "config": {
          "widget_type": "APPS",
          "visualization": "TABLE",
          "datasource": "TRAFFIC"
        },
        "widget_id": 1
      },
      {
        "title": "VoIP-RTP: Traffic Quality",
        "timestamp": "1383141976.674428",
        "criteria": {
          "traffic_expression": "",
          "sort_desc": true,
          "centricity": "host",
          "columns": [
            803
          ],
        },
        "time_frame": {
          "data_resolution": "min",
          "type": "last_hour",
          "refresh_interval": "min"
        },
        "attributes": {
          "format_bytes": "UI_PREF",
          "colspan": 1,
          "extend_to_zero": false,
          "line_scale": "LINEAR",
          "line_style": "STACKED"
        },
        "config": {
          "widget_type": "TRAFFIC_OVERALL",
          "visualization": "LINE",
          "datasource": "TRAFFIC"
        },
        "widget_id": 2
      },
      {
        "title": "VoIP-RTP: Traffic Quality",
        "timestamp": "1383141976.674459",
        "criteria": {
          "traffic_expression": "",
```



```
"sort_desc": true,
"centricity": "host",
"columns": [
  781
],
"time_frame": {
  "data_resolution": "min",
  "type": "last_hour",
  "refresh_interval": "min"
}
},
"attributes": {
  "format_bytes": "UI_PREF",
  "colspan": 1,
  "extend_to_zero": false,
  "line_style": "STACKED"
},
"config": {
  "widget_type": "TRAFFIC_OVERALL",
  "visualization": "LINE",
  "datasource": "TRAFFIC"
},
"widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      33
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_day",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": ""
```

```

"traffic_expression": "",
"host_group_type": "ByLocation",
"sort_desc": true,
"centricity": "host",
"limit": 100,
"time_frame": {
  "data_resolution": "15mins",
  "type": "last_hour",
  "refresh_interval": "15mins"
}
},
"attributes": {
  "format_bytes": "UI_PREF",
  "show_images": true,
  "layout": "HORIZONTAL_TREE",
  "colspan": 2,
  "moveable_nodes": true,
  "height": 400,
  "edge_thickness": true,
  "pan_zoomable": true,
  "n_items": 10
},
"config": {
  "widget_type": "HOST_GROUP_PAIRS",
  "visualization": "CONN_GRAPH",
  "datasource": "TRAFFIC"
},
"widget_id": 6
},
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      },
      {
        "id": 3
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 4
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 5
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 6
      }
    ]
  }
],
"section_id": 1
}
],
"id": 5217,
"description": ""
}

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpec</i>	<object>	Reporting template specification object.	
<i>ReportTemplateSpec.traffic_expression</i>	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpec.id</i>	<number>	ID of the report template.	Optional
<i>ReportTemplateSpec.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional

<i>ReportTemplateSpec.sharing</i>	<object>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplateSpec.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpec.sharing.users[item]</i>	<number>	User ID.	Optional
<i>ReportTemplateSpec.layout</i>	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.description</i>	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpec.user_id</i>	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpec.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpec.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpec.last_added_section_id</i>	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpec.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplateSpec.last_added_widget_id</i>	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpec.version</i>	<string>	Version of the specification.	Optional
<i>ReportTemplateSpec.disabled</i>	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpec.timestamp</i>	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpec.sections</i>	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpec.sections[TMSection]</i>	<object>	One TMSection object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget]</i>	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.ports [CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort]</i>	<object>	One CDSCAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService]</i>	<object>	One CService object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services [CService].service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns [item]</i>	<number>	Column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer]</i>	<object>	One CAApplicationServer object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].app</i>	<object>	Application specification.	

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server</i>	<object>	Server specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers [CAApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice]</i>	<object>	One CDevice object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices [CDevice].name</i>	<string>	Device name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort]</i>	<object>	One CAApplicationPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CAApplicationPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort]</i>	<object>	One CHostPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface]</i>	<object>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface</i>	<object>	Interface specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp</i>	<object>	DSCP specification.	



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service</i>	<object>	Watched service.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location</i>	<object>	Watched service location.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser]</i>	<object>	One CUser object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser].name</i>	<string>	Active Directory user name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_column</i>	<number>	Sorting column ID.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]</i>	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<object>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<object>	Segment source.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<object>	Segment destination.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<string>	Interface name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]</i>	<i>&lt;object&gt;</i>	One CHost object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair]</i>	<i>&lt;object&gt;</i>	One CHostPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server</i>	<i>&lt;object&gt;</i>	Specification of the server host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client</i>	<i>&lt;object&gt;</i>	Specification of the client host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched protocols.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol]</i>	<i>&lt;object&gt;</i>	Object representing Protocol information.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].id</i>	<i>&lt;number&gt;</i>	ID of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].name</i>	<i>&lt;string&gt;</i>	Name of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.centricity</i>	<i>&lt;string&gt;</i>	Centricity used to run the report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.limit</i>	<i>&lt;number&gt;</i>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface]</i>	<i>&lt;object&gt;</i>	One CInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [Interface].ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched host groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup]</i>	<i>&lt;object&gt;</i>	One CHostGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].name</i>	<i>&lt;string&gt;</i>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].group_id</i>	<i>&lt;number&gt;</i>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched DSCPs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP]</i>	<i>&lt;object&gt;</i>	One CDSCP object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].name</i>	<i>&lt;string&gt;</i>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].code_point</i>	<i>&lt;number&gt;</i>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched applications.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication]</i>	<i>&lt;object&gt;</i>	One CAApplication object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].code</i>	<i>&lt;string&gt;</i>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].name</i>	<i>&lt;string&gt;</i>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].tunneled</i>	<i>&lt;string&gt;</i>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].title</i>	<i>&lt;string&gt;</i>	Widget title.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes</i>	<i>&lt;object&gt;</i>	Widget common attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.modal_links</i>	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].timestamp</i>	<string>	Widget time stamp specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].section_id</i>	<number>	Section ID.	
<i>ReportTemplateSpec.sections[TMSection].layout</i>	<array of <object>>	Internal section layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.img</i>	<object>	Images associated with the template.	Optional
<i>ReportTemplateSpec.img.thumbnail</i>	<object>	A thumbnail-size image for the report template.	Optional
<i>ReportTemplateSpec.img.thumbnail.src</i>	<string>	Relative URL of an image.	
<i>ReportTemplateSpec.img.full</i>	<object>	A full-size image for the report template.	Optional
<i>ReportTemplateSpec.img.full.src</i>	<string>	Relative URL of an image.	

## Reporting: Clone template

Clone the specified reporting template.

POST <https://{device}/api/profiler/1.2/reporting/templates/copy?name={string}>

## Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>template</i>	<i>&lt;number&gt;</i>	ID of the template being cloned.	
<i>name</i>	<i>&lt;string&gt;</i>	A new unique name for the copy.	Optional

## Request Body

Do not provide a request body.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "traffic_expression": string,
  "schedule_type": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "name": string,
  "disabled": string,
  "next_run": number
}
```

Example:

```
{
  "user_id": 1,
  "live": true,
  "id": 1000,
  "name": "My Template"
}
```

Property Name	Type	Description	Notes
<i>ReportTemplate</i>	<i>&lt;object&gt;</i>	A template for running reports.	
<i>ReportTemplate.traffic_expression</i>	<i>&lt;string&gt;</i>	Traffic expression applied to all widgets within this template.	Optional
<i>ReportTemplate.schedule_type</i>	<i>&lt;string&gt;</i>	Type of template scheduling.	Optional; Values: Once, Hourly, Daily, Weekly, Monthly, Quarterly
<i>ReportTemplate.id</i>	<i>&lt;number&gt;</i>	ID of the template.	
<i>ReportTemplate.scheduled</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplate.sharing</i>	<i>&lt;object&gt;</i>	List of the users the template is shared with (see <i>ReportTemplateSharing</i> ).	Optional
<i>ReportTemplate.sharing.users</i>	<i>&lt;array of &lt;number&gt;&gt;</i>	List of the users a template is shared with.	Optional
<i>ReportTemplate.sharing.users[item]</i>	<i>&lt;number&gt;</i>	User ID.	Optional
<i>ReportTemplate.description</i>	<i>&lt;string&gt;</i>	Description of the template.	Optional
<i>ReportTemplate.user_id</i>	<i>&lt;number&gt;</i>	ID of the user who owns the template.	Optional
<i>ReportTemplate.shared</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplate.live</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is a dashboard.	
<i>ReportTemplate.name</i>	<i>&lt;string&gt;</i>	Human-readable name of the template.	
<i>ReportTemplate.disabled</i>	<i>&lt;string&gt;</i>	Flag indicating that data collection for the template is disabled.	Optional
<i>ReportTemplate.next_run</i>	<i>&lt;number&gt;</i>	Next run time for the template if the template is scheduled to run.	Optional

## Reporting: Delete template

Delete reporting template.



DELETE https://{device}/api/profiler/1.2/reporting/templates/{template\_id}

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "traffic_expression": string,
  "schedule_type": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "description": string,
  "user_id": number,
  "shared": string,
  "live": string,
  "name": string,
  "disabled": string,
  "next_run": number
}
```

Example:

```
{
  "user_id": 1,
  "live": true,
  "id": 1000,
  "name": "My Template"
}
```

Property Name	Type	Description	Notes
<i>ReportTemplate</i>	<i>&lt;object&gt;</i>	A template for running reports.	
<i>ReportTemplate.traffic_expression</i>	<i>&lt;string&gt;</i>	Traffic expression applied to all widgets within this template.	Optional
<i>ReportTemplate.schedule_type</i>	<i>&lt;string&gt;</i>	Type of template scheduling.	Optional; Values: Once, Hourly, Daily, Weekly, Monthly, Quarterly
<i>ReportTemplate.id</i>	<i>&lt;number&gt;</i>	ID of the template.	
<i>ReportTemplate.scheduled</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplate.sharing</i>	<i>&lt;object&gt;</i>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplate.sharing.users</i>	<i>&lt;array of &lt;number&gt;&gt;</i>	List of the users a template is shared with.	Optional
<i>ReportTemplate.sharing.users[item]</i>	<i>&lt;number&gt;</i>	User ID.	Optional
<i>ReportTemplate.description</i>	<i>&lt;string&gt;</i>	Description of the template.	Optional
<i>ReportTemplate.user_id</i>	<i>&lt;number&gt;</i>	ID of the user who owns the template.	Optional
<i>ReportTemplate.shared</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplate.live</i>	<i>&lt;string&gt;</i>	Flag indicating that the template is a dashboard.	
<i>ReportTemplate.name</i>	<i>&lt;string&gt;</i>	Human-readable name of the template.	
<i>ReportTemplate.disabled</i>	<i>&lt;string&gt;</i>	Flag indicating that data collection for the template is disabled.	Optional
<i>ReportTemplate.next_run</i>	<i>&lt;number&gt;</i>	Next run time for the template if the template is scheduled to run.	Optional

## Reporting: List templates

Get a list of templates.

GET https://{device}/api/profiler/1.2/reporting/templates?offset={number}&access={string}&filter={string}&live={string}&limit={number}

## Authorization



This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<number>	Starting element number.	Optional
<i>access</i>	<string>	Get only template with the specified access level: 'public' or 'private'.	Optional
<i>filter</i>	<string>	Apply a named filter: 'owned' to get only templates owned by the current user.	Optional
<i>live</i>	<string>	Filter only live (dashboard) templates.	Optional
<i>limit</i>	<number>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
[
  {
    "traffic_expression": string,
    "schedule_type": string,
    "id": number,
    "scheduled": string,
    "sharing": {
      "users": [
        number
      ]
    },
    "description": string,
    "user_id": number,
    "shared": string,
    "live": string,
    "name": string,
    "disabled": string,
    "next_run": number
  }
]
```

Example:

```
[
  {
    "scheduled": false,
    "live": false,
    "id": 184,
    "name": "Default template for class QUERY"
  },
  {
    "scheduled": true,
    "next_run": 1352328480,
    "user_id": 1,
    "live": true,
    "schedule_type": "Hourly",
    "id": 1000,
    "name": "My Template"
  }
]
```

Property Name	Type	Description	Notes
<i>ReportTemplates</i>	<array of <object>>	List of templates available on the system.	
<i>ReportTemplates[ReportTemplate]</i>	<object>	One template in the list of templates.	Optional
<i>ReportTemplates[ReportTemplate].traffic_expression</i>	<string>	Traffic expression applied to all widgets within this template.	Optional
<i>ReportTemplates[ReportTemplate].schedule_type</i>	<string>	Type of template scheduling.	Optional; Values: Once, Hourly, Daily, Weekly, Monthly, Quarterly
<i>ReportTemplates[ReportTemplate].id</i>	<number>	ID of the template.	
<i>ReportTemplates[ReportTemplate].scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplates[ReportTemplate].sharing</i>	<object>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplates[ReportTemplate].sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplates[ReportTemplate].sharing.users[item]</i>	<number>	User ID.	Optional

<code>ReportTemplates[ReportTemplate].description</code>	<code>&lt;string&gt;</code>	Description of the template.	Optional
<code>ReportTemplates[ReportTemplate].user_id</code>	<code>&lt;number&gt;</code>	ID of the user who owns the template.	Optional
<code>ReportTemplates[ReportTemplate].shared</code>	<code>&lt;string&gt;</code>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<code>ReportTemplates[ReportTemplate].live</code>	<code>&lt;string&gt;</code>	Flag indicating that the template is a dashboard.	
<code>ReportTemplates[ReportTemplate].name</code>	<code>&lt;string&gt;</code>	Human-readable name of the template.	
<code>ReportTemplates[ReportTemplate].disabled</code>	<code>&lt;string&gt;</code>	Flag indicating that data collection for the template is disabled.	Optional
<code>ReportTemplates[ReportTemplate].next_run</code>	<code>&lt;number&gt;</code>	Next run time for the template if the template is scheduled to run.	Optional

## Reporting: Get factory widgets

Get a list of all available stock widget configurations.

GET <https://{{device}}/api/profiler/1.2/reporting/templates/widgets>

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "config": {
      "datasource": string,
      "visualization": string,
      "widget_type": string
    },
    "widget_id": number,
    "criteria": {
      "ports": [
        {
          "port": number,
          "protocol": number,
          "name": string
        }
      ],
      "dscp_app_ports": [
        {
          "port": {
            "port": number,
            "protocol": number,
            "name": string
          },
          "app": {
            "code": string,
            "name": string,
            "tunneled": string
          },
          "dscp": {
            "name": string,
            "code_point": number
          }
        }
      ],
      "services": [
        {
          "name": string,
          "service_id": number
        }
      ],
      "port_groups": [
        {
          "name": string,
          "group_id": number
        }
      ],
      "comparison_time_frame": {
        "data_resolution": string,
        "refresh_interval": string,
        "type": string
      }
    }
  }
]
```

```
,
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
```

```

"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
]

```

```
    }
  ],
  "hosts": [
    {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  ],
  "host_pairs": [
    {
      "server": {
        "mac": string,
        "ipaddr": string,
        "name": string
      },
      "client": {
        "mac": string,
        "ipaddr": string,
        "name": string
      }
    }
  ],
  "protocols": [
    {
      "id": number,
      "name": string
    }
  ],
  "centricity": string,
  "limit": number,
  "interfaces": [
    {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  ],
  "host_groups": [
    {
      "name": string,
      "group_id": number
    }
  ],
  "dscps": [
    {
      "name": string,
      "code_point": number
    }
  ],
  "applications": [
    {
      "code": string,
      "name": string,
      "tunneled": string
    }
  ]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
```

```

"format_bytes": string,
"show_images": string,
"open_nodes": [
  string
],
"line_style": string,
"layout": string,
"width": number,
"height": number,
"percent_of_total": string,
"edge_thickness": string,
"display_host_group_type": string,
"extend_to_zero": string,
"collapsible": string,
"high_threshold": string,
"n_items": number,
"colspan": number,
"low_threshold": string,
"moveable_nodes": string,
"orientation": string,
"modal_links": number
},
"timestamp": string
}
]

```

Example:

```

[
{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}
]

```

Property Name	Type	Description	Notes
<i>TMWidgets</i>	<array of <object>>	List of TMWidget objects.	
<i>TMWidgets</i> [TMWidget]	<object>	One TMWidget object.	Optional
<i>TMWidgets</i> [TMWidget].config	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidgets</i> [TMWidget].config.datasource	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidgets</i> [TMWidget].config.visualization	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE

<i>TMWidgets</i> [ <i>TMWidget</i> ].config.widget_type	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidgets</i> [ <i>TMWidget</i> ].widget_id	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria	<object>	Query criteria for the widget.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.ports	<array of <object>>	Watched ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.ports [ <i>CProtoPort</i> ]	<object>	One <i>CProtoPort</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.ports [ <i>CProtoPort</i> ].port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.ports [ <i>CProtoPort</i> ].protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.ports [ <i>CProtoPort</i> ].name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ]	<object>	One <i>CDSCAppPort</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].port. protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].app. tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].dscp	<object>	DSCP specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].dscp.name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. dscp_app_ports[ <i>CDSCAppPort</i> ].dscp. code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.services	<array of <object>>	Watched services.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.services [ <i>CService</i> ]	<object>	One <i>CService</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.services [ <i>CService</i> ].name	<string>	Service name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.services [ <i>CService</i> ].service_id	<number>	Service ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.port_groups	<array of <object>>	Watched port groups.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.port_groups [ <i>CPortGroup</i> ]	<object>	One <i>CPortGroup</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.port_groups [ <i>CPortGroup</i> ].name	<string>	Name of the port group.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.port_groups [ <i>CPortGroup</i> ].group_id	<number>	ID of the port group.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.comparison_time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs	<array of <object>>	Watched group pairs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair]	<object>	One CHostGroupPair object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server	<object>	Server host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client	<object>	Client host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pairs[CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.wan_group	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.traffic_expression	<string>	Traffic expression.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.split_direction	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_successes	<string>	Include successful requests in active directory report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.columns	<array of <number>>	List of column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.columns[item]	<number>	Column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server	<object>	Server specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers[CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional



<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_servers [CApplicationServer].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice]	<object>	One CDevice object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice].ipaddr	<string>	Device IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.devices [CDevice].name	<string>	Device name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort]	<object>	One CApplicationPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.application_ports[CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server	<object>	Server host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].client	<object>	Client host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_ports[CHostPairPort].client.mac	<string>	Host MAC address.	Optional

<i>TMWidgets</i> [TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [TMWidget].criteria.host_pair_ports[CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [TMWidget].criteria.time_frame	<object>	Widget time frame specification.	Optional
<i>TMWidgets</i> [TMWidget].criteria.time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [TMWidget].criteria.time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidgets</i> [TMWidget].criteria.time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidgets</i> [TMWidget].criteria.service	<object>	Watched service.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service.name	<string>	Service name.	
<i>TMWidgets</i> [TMWidget].criteria.service.service_id	<number>	Service ID.	Optional
<i>TMWidgets</i> [TMWidget].criteria.severity	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidgets</i> [TMWidget].criteria.role	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidgets</i> [TMWidget].criteria.event_policies	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidgets</i> [TMWidget].criteria.event_policies[item]	<number>	Event policy ID.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service_locations	<array of <object>>	Watched service locations.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service_locations[CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service_locations[CServiceLocation].name	<string>	Service location name.	
<i>TMWidgets</i> [TMWidget].criteria.service_locations[CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>TMWidgets</i> [TMWidget].criteria.case_insensitive	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service_location	<object>	Watched service location.	Optional
<i>TMWidgets</i> [TMWidget].criteria.service_location.name	<string>	Service location name.	
<i>TMWidgets</i> [TMWidget].criteria.service_location.location_id	<string>	Service location ID.	Optional
<i>TMWidgets</i> [TMWidget].criteria.include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_type	<string>	Host group type used.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ]	<object>	One <i>CHostPairAppPort</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].port.port	<number>	Port specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].app	<object>	Application specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].app.code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].app.name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].server	<object>	Server host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].server.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].client	<object>	Client host specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].client.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pair_app_ports[ <i>CHostPairAppPort</i> ].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users	<array of <object>>	Watched users.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users[ <i>CUser</i> ]	<object>	One <i>CUser</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.users[ <i>CUser</i> ].name	<string>	Active Directory user name.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.sort_desc	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.sort_column	<number>	Sorting column ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports[ <i>CHostGroupPairPort</i> ]	<object>	One <i>CHostGroupPairPort</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports[ <i>CHostGroupPairPort</i> ].port	<object>	Port specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_group_pair_ports[ <i>CHostGroupPairPort</i> ].port.port	<number>	Port specification.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server	<object>	Server host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].server.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client	<object>	Client host group specification.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client.name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. host_group_pair_ports [ <i>CHostGroupPairPort</i> ].client.group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments	<array of <object>>	Watched network segments.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ]	<object>	One <i>CNetworkSegment</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src	<object>	Segment source.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].src. ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst	<object>	Segment destination.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria. network_segments[ <i>CNetworkSegment</i> ].dst. ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts	<array of <object>>	Watched hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ]	<object>	One <i>CHost</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.hosts [ <i>CHost</i> ].name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs	<array of <object>>	Watched host pairs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ]	<object>	One <i>CHostPair</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server	<object>	Specification of the server host.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [ <i>CHostPair</i> ].server.name	<string>	Host name.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client	<object>	Specification of the client host.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.mac	<string>	Host MAC address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_pairs [C <i>HostPair</i> ].client.name	<string>	Host name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ]	<object>	Object representing Protocol information.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ].id	<number>	ID of the Protocol.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.protocols [C <i>Protocol</i> ].name	<string>	Name of the Protocol.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces	<array of <object>>	Watched interfaces.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ]	<object>	One C <i>Interface</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].ipaddr	<string>	Interface IP address.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].name	<string>	Interface name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.interfaces [C <i>Interface</i> ].ifindex	<number>	Interface index.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups	<array of <object>>	Watched host groups.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ]	<object>	One C <i>HostGroup</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ].name	<string>	Host group name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.host_groups [C <i>HostGroup</i> ].group_id	<number>	Host group ID.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ]	<object>	One C <i>CDSCP</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ].name	<string>	DSCP name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.dscps [C <i>CDSCP</i> ].code_point	<number>	DSCP code point.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications	<array of <object>>	Watched applications.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications [C <i>Application</i> ]	<object>	One C <i>Application</i> object.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications [C <i>Application</i> ].code	<string>	Application code.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications [C <i>Application</i> ].name	<string>	Application name.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].criteria.applications [C <i>Application</i> ].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].title	<string>	Widget title.	
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes	<object>	Widget common attributes.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.width	<number>	Widget width.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.height	<number>	Widget height.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgets</i> [ <i>TMWidget</i> ].attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes	<object>	User-specific attributes.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.width	<number>	Widget width.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.height	<number>	Widget height.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional

<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgets</i> [ <i>TMWidget</i> ].user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMWidgets</i> [ <i>TMWidget</i> ].timestamp	<string>	Widget time stamp specification.	Optional

## Reporting: Export templates

Export all reporting templates.

GET <https://{device}/api/profiler/1.2/reporting/templates/export>

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "traffic_expression": string,
    "id": number,
    "scheduled": string,
    "sharing": {
      "users": [
        number
      ]
    },
    "layout": [
      {
        "flow_items": [
          {
            "id": number
          }
        ],
        "attributes": {
          "wrappable": string,
          "full_width": string,
          "item_spacing": string
        }
      }
    ],
    "description": string,
    "user_id": number,
    "shared": string,
    "live": string,
    "last_added_section_id": number,
    "name": string,
    "last_added_widget_id": number,
    "version": string,
    "disabled": string,
    "timestamp": string,
    "sections": [
      {
        "widgets": [
          {
            "config": {
              "datasource": string,
              "visualization": string,
              "widget_type": string
            },
            "widget_id": number,
            "criteria": {
              "ports": [
                {
                  "port": number,
                  "protocol": number,
```

```
"name": string
},
],
"dscp_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"services": [
  {
    "name": string,
    "service_id": number
  }
],
"port_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"comparison_time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string
    }
  }
]
```



```
    name : string,
    "tunneled": string
  }
},
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
]
```

```
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
```

```

"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
],
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
},
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}

```

```
}  
]
```

Example:

```
[  
  {  
    "layout": [  
      {  
        "flow_items": [  
          {  
            "id": 1  
          }  
        ]  
      }  
    ],  
    "name": "VOIP - Call Quality and Usage",  
    "user_id": 1,  
    "timestamp": "1383141976.674345",  
    "live": true,  
    "last_added_widget_id": 6,  
    "traffic_expression": "app VoIP-RTP",  
    "version": "1.1",  
    "shared": "Private",  
    "sections": [  
      {  
        "widgets": [  
          {  
            "title": "VoIP-RTP: Applications",  
            "timestamp": "1383141976.674383",  
            "criteria": {  
              "sort_column": 33,  
              "traffic_expression": "",  
              "sort_desc": true,  
              "centricity": "host",  
              "limit": 100,  
              "columns": [  
                17,  
                33,  
                34,  
                757,  
                766,  
                781,  
                803  
              ],  
              "time_frame": {  
                "data_resolution": "15mins",  
                "type": "last_hour",  
                "refresh_interval": "15mins"  
              }  
            },  
            "attributes": {  
              "format_bytes": "UI_PREF",  
              "colspan": 2,  
              "n_items": 20  
            },  
            "config": {  
              "widget_type": "APPS",  
              "visualization": "TABLE",  
              "datasource": "TRAFFIC"  
            },  
            "widget_id": 1  
          },  
          {  
            "title": "VoIP-RTP: Traffic Quality",  
            "timestamp": "1383141976.674428",  
            "criteria": {  
              "traffic_expression": "",  
              "sort_desc": true,  
              "centricity": "host",  
              "columns": [  
                803  
              ],  
              "time_frame": {  
                "data_resolution": "min",  
                "type": "last_hour",  
                "refresh_interval": "min"  
              }  
            },  
            "attributes": {  
              "format_bytes": "UI_PREF",  
              "colspan": 1,  
              "extend_to_zero": false,  
              "line_scale": "LINEAR",  
              "line_style": "STACKED"  
            },  
            "config": {  
              "widget_type": "TRAFFIC_OVERALL",  
                "format_bytes": "UI_PREF",  
                "colspan": 1,  
                "extend_to_zero": false,  
                "line_scale": "LINEAR",  
                "line_style": "STACKED"  
            }  
          }  
        ]  
      }  
    ]  
  }  
]
```

```
"visualization": "LINE",
"datasource": "TRAFFIC"
},
"widget_id": 2
},
{
"title": "VoIP-RTP: Traffic Quality",
"timestamp": "1383141976.674459",
"criteria": {
"traffic_expression": "",
"sort_desc": true,
"centricity": "host",
"columns": [
781
],
"time_frame": {
"data_resolution": "min",
"type": "last_hour",
"refresh_interval": "min"
}
},
"attributes": {
"format_bytes": "UI_PREF",
"colspan": 1,
"extend_to_zero": false,
"line_style": "STACKED"
},
"config": {
"widget_type": "TRAFFIC_OVERALL",
"visualization": "LINE",
"datasource": "TRAFFIC"
},
"widget_id": 3
},
{
"title": "VoIP-RTP: Traffic Quality",
"timestamp": "1383141976.674497",
"criteria": {
"traffic_expression": "",
"sort_desc": true,
"centricity": "host",
"columns": [
766
],
"time_frame": {
"data_resolution": "min",
"type": "last_hour",
"refresh_interval": "min"
}
},
"attributes": {
"format_bytes": "UI_PREF",
"colspan": 2,
"extend_to_zero": false,
"line_style": "STACKED"
},
"config": {
"widget_type": "TRAFFIC_OVERALL",
"visualization": "LINE",
"datasource": "TRAFFIC"
},
"widget_id": 4
},
{
"title": "VoIP-RTP: Traffic Volume",
"timestamp": "1383141976.674527",
"criteria": {
"traffic_expression": "",
"sort_desc": true,
"centricity": "host",
"columns": [
33
],
"time_frame": {
"data_resolution": "15mins",
"type": "last_day",
"refresh_interval": "15mins"
}
},
"attributes": {
"format_bytes": "UI_PREF",
"colspan": 2,
"extend_to_zero": false,
"line_style": "STACKED"
},
"config": {
"widget_type": "TRAFFIC_OVERALL",
"visualization": "LINE"
```

```

        visualization: LINE ,
        "datasource": "TRAFFIC"
    },
    "widget_id": 5
},
{
    "title": "Host Group Pairs",
    "timestamp": "1383141976.674566",
    "criteria": {
        "sort_column": 33,
        "traffic_expression": "",
        "host_group_type": "ByLocation",
        "sort_desc": true,
        "centricity": "host",
        "limit": 100,
        "time_frame": {
            "data_resolution": "15mins",
            "type": "last_hour",
            "refresh_interval": "15mins"
        }
    },
    "attributes": {
        "format_bytes": "UI_PREF",
        "show_images": true,
        "layout": "HORIZONTAL_TREE",
        "colspan": 2,
        "moveable_nodes": true,
        "height": 400,
        "edge_thickness": true,
        "pan_zoomable": true,
        "n_items": 10
    },
    "config": {
        "widget_type": "HOST_GROUP_PAIRS",
        "visualization": "CONN_GRAPH",
        "datasource": "TRAFFIC"
    },
    "widget_id": 6
}
],
"layout": [
    {
        "flow_items": [
            {
                "id": 1
            }
        ]
    },
    {
        "flow_items": [
            {
                "id": 2
            },
            {
                "id": 3
            }
        ]
    },
    {
        "flow_items": [
            {
                "id": 4
            }
        ]
    },
    {
        "flow_items": [
            {
                "id": 5
            }
        ]
    },
    {
        "flow_items": [
            {
                "id": 6
            }
        ]
    }
],
"section_id": 1
}
],
"id": 5217,
"description": ""
}
]

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpecs</i>	<array of <object>>	List of ReportTemplateSpec objects.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec]	<object>	One ReportTemplateSpecs object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].traffic_expression	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].id	<number>	ID of the report template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].scheduled	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing	<object>	List of the users the template is shared with (see ReportTemplateSharing).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing.users	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sharing.users[item]	<number>	User ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine]	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items [TMFlowItem]	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].flow_items [TMFlowItem].id	<number>	Widget ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes	<object>	List of line attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].layout[TMFlowLine].attributes.item_spacing	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].description	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].user_id	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].shared	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].live	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].last_added_section_id	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].name	<string>	Human-readable name of the template.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].last_added_widget_id	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].version	<string>	Version of the specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].disabled	<string>	Flag indicating that the template is disabled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].timestamp	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection]	<object>	One TMSection object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget]	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config	<object>	Widget configuration: data source type, widget type, and visualization type.	

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.datasources	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.visualization	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].config.widget_type	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].widget_id	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria	<object>	Query criteria for the widget.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort]	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.ports[CProtoPort].name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort]	<object>	One CDSCPAppPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.name	<string>	Application name.	Optional



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp	<object>	DSCP specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService]	<object>	One CService object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService].name	<string>	Service name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.services[CService].service_id	<number>	Service ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup]	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].name	<string>	Name of the port group.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].group_id	<number>	ID of the port group.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair]	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server	<object>	Server host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.group_id	<number>	Host group ID.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client	<object>	Client host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.wan_group	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.traffic_expression	<string>	Traffic expression.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.split_direction	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_successes	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.columns	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.columns[item]	<number>	Column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server	<object>	Server specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice]	<object>	One CDevice object.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].ipaddr	<string>	Device IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].name	<string>	Device name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort]	<object>	One CApplicationPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].app	<object>	Application specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort].app.tunneled	<string>	Flag; is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server	<object>	Server host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.mac	<string>	Host MAC address.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client	<object>	Client host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_ports [CHostPairPort].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface	<object>	Interface specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame	<object>	Widget time frame specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service	<object>	Watched service.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service.name	<string>	Service name.	

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service.service_id	<number>	Service ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.severity	<number>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.role	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.event_policies	<array of <number>>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.event_policies[item]	<number>	Event policy ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation].name	<string>	Service location name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_locations [CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.case_insensitive	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location	<object>	Watched service location.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location.name	<string>	Service location name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.service_location.location_id	<string>	Service location ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_type	<string>	Host group type used.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app	<object>	Application specification.	

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server	<object>	Server host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client	<object>	Client host specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users[CUser]	<object>	One CUser object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.users[CUser].name	<string>	Active Directory user name.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.sort_desc	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.sort_column	<number>	Sorting column ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port	<object>	Port specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments	<array of <object>>	Watched network segments.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src	<object>	Segment source.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst	<object>	Segment destination.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.network_segments [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts	<array of <object>>	Watched hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]	<object>	One CHost object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr	<string>	Host IP address.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.hosts[CHost].name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs	<array of <object>>	Watched host pairs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair]	<object>	One CHostPair object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server	<object>	Specification of the server host.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].server.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client	<object>	Specification of the client host.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.mac	<string>	Host MAC address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.ipaddr	<string>	Host IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_pairs[CHostPair].client.name	<string>	Host name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol]	<object>	Object representing Protocol information.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol].id	<number>	ID of the Protocol.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.protocols[CProtocol].name	<string>	Name of the Protocol.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces	<array of <object>>	Watched interfaces.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface]	<object>	One CInterface object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].ipaddr	<string>	Interface IP address.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].name	<string>	Interface name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.interfaces[CInterface].ifindex	<number>	Interface index.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups	<array of <object>>	Watched host groups.	Optional



<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup]	<object>	One CHostGroup object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup].name	<string>	Host group name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.host_groups[CHostGroup].group_id	<number>	Host group ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP]	<object>	One CDSCP object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].name	<string>	DSCP name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.dscps[CDSCP].code_point	<number>	DSCP code point.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications	<array of <object>>	Watched applications.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication]	<object>	One CAApplication object.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].code	<string>	Application code.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].name	<string>	Application name.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].criteria.applications[CAApplication].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].title	<string>	Widget title.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes	<object>	Widget common attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.width	<number>	Widget width.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.height	<number>	Widget height.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes	<object>	User-specific attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.width	<number>	Widget width.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.height	<number>	Widget height.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional

<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].widgets[TMWidget].timestamp	<string>	Widget time stamp specification.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].section_id	<number>	Section ID.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout	<array of <object>>	Internal section layout.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine]	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem]	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id	<number>	Widget ID.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes	<object>	List of line attributes.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].sections[TMSection].layout[TMFlowLine].attributes.item_spacing	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img	<object>	Images associated with the template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.thumbnail	<object>	A thumbnail-size image for the report template.	Optional
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.thumbnail.src	<string>	Relative URL of an image.	
<i>ReportTemplateSpecs</i> [ReportTemplateSpec].img.full	<object>	A full-size image for the report template.	Optional

ReportTemplateSpecs[ReportTemplateSpec].  
img.full.src

<string>

Relative URL of an image.

## Reporting: Get report config

Get configuration information for one report. Includes criteria, layout and GUI attributes.

GET https://{device}/api/profiler/1.2/reporting/reports/{report\_id}/config

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "criteria": {
    "traffic_expression": string,
    "time_frame": {
      "resolution": string,
      "end": number,
      "expression": string,
      "start": number
    },
    "query": {
      "columns": [
        number
      ],
      "role": string,
      "group_by": string,
      "host_group_type": string,
      "direction": string,
      "sort_column": number,
      "area": string,
      "centricity": string,
      "realm": string
    },
    "deprecated": {
      [prop]: string
    }
  },
  "attributes": {
    [prop]: string
  },
  "sections": [
    {
      "widgets": [
        {
          "query_id": string,
          "id": string,
          "type": string,
          "attributes": {
            [prop]: string
          }
        }
      ],
      "layout": [
        {
          "items": [
            {
              "id": string
            }
          ],
          "wrappable": string,
          "full_width": string,
          "item_spacing": string,
          "line_spacing": string
        }
      ],
      "attributes": {
        [prop]: string
      }
    }
  ]
}
```

Example:

```
{
```

```

"attributes": {
  "title": "Report name"
},
"sections": [
  {
    "widgets": [
      {
        "type": "table",
        "query_id": "0:sum_hos_non_non_non_non_non_non_non_33_d_0",
        "id": "sum_hos_non_non_non_non_non_non_non_tbl_0_0",
        "attributes": {
          "page_size": "20",
          "sort_col": "33",
          "col_order": "6,33,34"
        }
      }
    ],
    "attributes": {
      "sort_desc": "1"
    },
    "layout": [
      {
        "items": [
          {
            "id": "sum_hos_non_non_non_non_non_non_non_tbl_0_0"
          }
        ]
      }
    ]
  }
],
"criteria": {
  "time_frame": {
    "start": 1352319891,
    "end": 1352320191
  },
  "query": {
    "realm": "traffic_summary",
    "sort_column": 33,
    "centricity": "hos",
    "group_by": "hos",
    "columns": [
      6,
      33,
      34
    ]
  }
}
}

```

Property Name	Type	Description	Notes
<i>ReportConfig</i>	<object>	Object representing report configuration.	
<i>ReportConfig.criteria</i>	<object>	Report criteria.	
<i>ReportConfig.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportConfig.criteria.time_frame</i>	<object>	Time frame object.	Optional
<i>ReportConfig.criteria.time_frame.resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month. If not specified a resolution will automatically be selected based on time frame of the report.	Optional
<i>ReportConfig.criteria.time_frame.end</i>	<number>	Report end time (unix time).	Optional
<i>ReportConfig.criteria.time_frame.expression</i>	<string>	Traffic expression.	Optional
<i>ReportConfig.criteria.time_frame.start</i>	<number>	Report start time (unix time).	Optional
<i>ReportConfig.criteria.query</i>	<object>	Query object.	Optional
<i>ReportConfig.criteria.query.columns</i>	<array of <number>>	Query columns. Can be many of GET /reporting/columns.	Optional
<i>ReportConfig.criteria.query.columns [item]</i>	<number>	Query column.	Optional
<i>ReportConfig.criteria.query.role</i>	<string>	Query role. Can be one of /reporting/roles.	Optional
<i>ReportConfig.criteria.query.group_by</i>	<string>	Query group_by. Can be one of GET /reporting/group_bys.	Optional
<i>ReportConfig.criteria.query.host_group_type</i>	<string>	Query host group type. Required for "host group (gro)" "host group pairs (gpp)" and "host group pairs with ports (gpr)" queries.	Optional
<i>ReportConfig.criteria.query.direction</i>	<string>	Query direction. Can be one of GET /reporting/directions.	Optional
<i>ReportConfig.criteria.query.sort_column</i>	<number>	Query sort column. Can be one of GET /reporting/columns.	Optional

<i>ReportConfig.criteria.query.area</i>	<i>&lt;string&gt;</i>	Query area. Can be one of GET /reporting/areas.	Optional
<i>ReportConfig.criteria.query.centricity</i>	<i>&lt;string&gt;</i>	Query centricity. Can be one of GET /reporting/centricities.	Optional
<i>ReportConfig.criteria.query.realm</i>	<i>&lt;string&gt;</i>	Query realm. Can be one of GET /reporting/realm.	
<i>ReportConfig.criteria.deprecated</i>	<i>&lt;object&gt;</i>	Map with legacy criteria attributes that will not be supported soon.	Optional
<i>ReportConfig.criteria.deprecated[prop]</i>	<i>&lt;string&gt;</i>	ReportDeprecatedFilters map value.	Optional
<i>ReportConfig.attributes</i>	<i>&lt;object&gt;</i>	Report attributes.	
<i>ReportConfig.attributes[prop]</i>	<i>&lt;string&gt;</i>	Report attributes value.	Optional
<i>ReportConfig.sections</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Report sections.	
<i>ReportConfig.sections[ReportSection]</i>	<i>&lt;object&gt;</i>	One section of a report.	Optional
<i>ReportConfig.sections[ReportSection].widgets</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of section widgets.	
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget]</i>	<i>&lt;object&gt;</i>	One widget from a list of widgets.	Optional
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget].query_id</i>	<i>&lt;string&gt;</i>	Query ID for the query that the widget is based on.	
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget].id</i>	<i>&lt;string&gt;</i>	Widget ID used to reference a widget from the API.	
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget].type</i>	<i>&lt;string&gt;</i>	Visual type of the widget.	
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget].attributes</i>	<i>&lt;object&gt;</i>	Widget attributes.	
<i>ReportConfig.sections[ReportSection].widgets[ReportWidget].attributes[prop]</i>	<i>&lt;string&gt;</i>	Attribute value in the map.	Optional
<i>ReportConfig.sections[ReportSection].layout</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Section widget layout.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine]</i>	<i>&lt;object&gt;</i>	One horizontal line of widgets.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].items</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of items (widgets) on the line.	
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].items[ReportLayoutItem]</i>	<i>&lt;object&gt;</i>	One item in the list of layout items.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].items[ReportLayoutItem].id</i>	<i>&lt;string&gt;</i>	ID of the layout item.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].wrappable</i>	<i>&lt;string&gt;</i>	Flag allowing wrapping.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].full_width</i>	<i>&lt;string&gt;</i>	Flag representing width of the layout line.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].item_spacing</i>	<i>&lt;string&gt;</i>	Item spacing between widgets.	Optional
<i>ReportConfig.sections[ReportSection].layout[ReportLayoutLine].line_spacing</i>	<i>&lt;string&gt;</i>	Line spacing.	Optional
<i>ReportConfig.sections[ReportSection].attributes</i>	<i>&lt;object&gt;</i>	Section attributes.	
<i>ReportConfig.sections[ReportSection].attributes[prop]</i>	<i>&lt;string&gt;</i>	Attribute value in the map.	Optional

## Reporting: List areas

Get a list of areas that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/areas
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "wan",
    "name": "wan"
  },
  {
    "id": "lan",
    "name": "lan"
  }
]
```

Property Name	Type	Description	Notes
Areas	<array of <object>>	List of areas.	
Areas[Area]	<object>	Object representing an area.	Optional
Areas[Area].id	<string>	ID of an area. To be used in the API.	
Areas[Area].name	<string>	Human-readable name of a area.	

## Reporting: Update user attributes

Merge user-specific template attributes with the new attribute values.

PUT [https://{device}/api/profiler/1.2/reporting/templates/{template\\_id}/sections/{section\\_id}/widgets/{widget\\_id}/user\\_attributes](https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}/widgets/{widget_id}/user_attributes)

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

#### JSON

```
{
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
}
```

Example:

```
[]
```

Property Name	Type	Description	Notes
TMWidgetAttributes	<object>	Set of widget attributes.	
TMWidgetAttributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional

<i>TMWidgetAttributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidgetAttributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidgetAttributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidgetAttributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidgetAttributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidgetAttributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidgetAttributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidgetAttributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidgetAttributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidgetAttributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidgetAttributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidgetAttributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidgetAttributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidgetAttributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidgetAttributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidgetAttributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidgetAttributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidgetAttributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidgetAttributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidgetAttributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidgetAttributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional

## Response Body

On success, the server does not provide any body in the responses.

## Reporting: List metrics

Get a list of metrics that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/metrics
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

```
JSON
```



```
[
  {
    "id": string,
    "name": string
  }
]

Example:
[
  {
    "id": "nbw",
    "name": "net bandwidth"
  },
  {
    "id": "nrt",
    "name": "net rtt"
  },
  {
    "id": "rtm",
    "name": "response time"
  }
]
```

Property Name	Type	Description	Notes
<i>Metrics</i>	<array of <object>>	List of metrics.	
<i>Metrics</i> [Metric]	<object>	Object representing a metric.	Optional
<i>Metrics</i> [Metric].id	<string>	ID of a metric. To be used in the API.	
<i>Metrics</i> [Metric].name	<string>	Human-readable name of a metric.	

## Reporting: List rates

Get a list of rates that this version of the API supports.

GET <https://{device}/api/profiler/1.2/reporting/rates>

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]

Example:
[
  {
    "id": "cnt",
    "name": "count"
  },
  {
    "id": "psc",
    "name": "per second"
  }
]
```

Property Name	Type	Description	Notes
<i>Rates</i>	<array of <object>>	List of rates.	
<i>Rates</i> [Rate]	<object>	Object representing a rate.	Optional
<i>Rates</i> [Rate].id	<string>	ID of a rate. To be used in the API.	
<i>Rates</i> [Rate].name	<string>	Human-readable name of a rate.	

## Reporting: Get template section

## Reporting: Get Template Section

Get the specific section from the template.

```
GET https://{device}/api/profiler/1.2/reporting/templates/{template_id}/sections/{section_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "widgets": [
    {
      "config": {
        "datasource": string,
        "visualization": string,
        "widget_type": string
      },
      "widget_id": number,
      "criteria": {
        "ports": [
          {
            "port": number,
            "protocol": number,
            "name": string
          }
        ],
        "dscp_app_ports": [
          {
            "port": {
              "port": number,
              "protocol": number,
              "name": string
            },
            "app": {
              "code": string,
              "name": string,
              "tunneled": string
            },
            "dscp": {
              "name": string,
              "code_point": number
            }
          }
        ],
        "services": [
          {
            "name": string,
            "service_id": number
          }
        ],
        "port_groups": [
          {
            "name": string,
            "group_id": number
          }
        ],
        "comparison_time_frame": {
          "data_resolution": string,
          "refresh_interval": string,
          "type": string
        },
        "host_group_pairs": [
          {
            "server": {
              "name": string,
              "group_id": number
            },
            "client": {
              "name": string,
              "group_id": number
            }
          }
        ],
        "wan_group": string,
        "traffic_expression": string,
        "split_direction": string,
        "include_successes": string,
        "include non optimized sites": string
      }
    }
  ]
}
```

```
include_non_optimized_apps : string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
```

```
"location_id": string
}
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
```

```
    "mac": string,
    "ipaddr": string,
    "name": string
  }
},
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
```

```

    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
}
}

```

Property Name	Type	Description	Notes
<i>TMSection</i>	<object>	One section in the report layout.	
<i>TMSection.widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>TMSection.widgets[TMWidget]</i>	<object>	One TMWidget object.	Optional
<i>TMSection.widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMSection.widgets[TMWidget].config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMSection.widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>TMSection.widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMSection.widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMSection.widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>TMSection.widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional

<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app</i>	<object>	Application specification.	
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_app_ports[CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMSection.widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMSection.widgets[TMWidget].criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMSection.widgets[TMWidget].criteria.services[CService].name</i>	<string>	Service name.	
<i>TMSection.widgets[TMWidget].criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMSection.widgets[TMWidget].criteria.port_groups[CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMSection.widgets[TMWidget].criteria.port_groups[CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>TMSection.widgets[TMWidget].criteria.port_groups[CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>TMSection.widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMSection.widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSection.widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSection.widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of 'WAN', 'WAN/Optimized', 'WAN/Non-optimized'.	Optional
<i>TMSection.widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMSection.widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional

<i>TMSection.widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMSection.widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMSection.widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_servers [CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMSection.widgets[TMWidget].criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>TMSection.widgets[TMWidget].criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].port</i>	<object>	Port specification.	
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].port.port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].app</i>	<object>	Application specification.	
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].app.code</i>	<string>	Application code.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].app.name</i>	<string>	Application name.	Optional
<i>TMSection.widgets[TMWidget].criteria.application_ports[CApplicationPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional



<i>TMSection.widgets[TMWidget].criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort]</i>	<object>	One CHostPairPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port</i>	<object>	Port specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface]</i>	<object>	One CDSCPInterface object.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface</i>	<object>	Interface specification.	
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name</i>	<string>	Interface name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex</i>	<number>	Interface index.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp</i>	<object>	DSCP specification.	
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMSection.widgets[TMWidget].criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSection.widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSection.widgets[TMWidget].criteria.time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month

<i>TMSection.widgets[TMWidget].criteria.service</i>	<object>	Watched service.	Optional
<i>TMSection.widgets[TMWidget].criteria.service.name</i>	<string>	Service name.	
<i>TMSection.widgets[TMWidget].criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMSection.widgets[TMWidget].criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMSection.widgets[TMWidget].criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMSection.widgets[TMWidget].criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMSection.widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>TMSection.widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<string>	Service location name.	
<i>TMSection.widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMSection.widgets[TMWidget].criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMSection.widgets[TMWidget].criteria.service_location.name</i>	<string>	Service location name.	
<i>TMSection.widgets[TMWidget].criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<object>	Port specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<object>	Application specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional

<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMSection.widgets[TMWidget].criteria.users[CUser]</i>	<object>	One CUser object.	Optional
<i>TMSection.widgets[TMWidget].criteria.users[CUser].name</i>	<string>	Active Directory user name.	
<i>TMSection.widgets[TMWidget].criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMSection.widgets[TMWidget].criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]</i>	<object>	One CHostGroupPairPort object.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port</i>	<object>	Port specification.	
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port</i>	<number>	Port specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server</i>	<object>	Server host group specification.	
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name</i>	<string>	Host group name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id</i>	<number>	Host group ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client</i>	<object>	Client host group specification.	
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name</i>	<string>	Host group name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id</i>	<number>	Host group ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<object>	One CNetworkSegment object.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<object>	Segment source.	
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<string>	Interface name.	Optional

<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<number>	Interface index.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<object>	Segment destination.	
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<string>	Interface name.	Optional
<i>TMSection.widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<number>	Interface index.	Optional
<i>TMSection.widgets[TMWidget].criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMSection.widgets[TMWidget].criteria.hosts[CHost]</i>	<object>	One CHost object.	Optional
<i>TMSection.widgets[TMWidget].criteria.hosts[CHost].mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.hosts[CHost].ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.hosts[CHost].name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair]</i>	<object>	One CHostPair object.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].server</i>	<object>	Specification of the server host.	
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].server.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].client</i>	<object>	Specification of the client host.	
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_pairs[CHostPair].client.name</i>	<string>	Host name.	Optional
<i>TMSection.widgets[TMWidget].criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMSection.widgets[TMWidget].criteria.protocols[CProtocol]</i>	<object>	Object representing Protocol information.	Optional
<i>TMSection.widgets[TMWidget].criteria.protocols[CProtocol].id</i>	<number>	ID of the Protocol.	Optional
<i>TMSection.widgets[TMWidget].criteria.protocols[CProtocol].name</i>	<string>	Name of the Protocol.	Optional
<i>TMSection.widgets[TMWidget].criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMSection.widgets[TMWidget].criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMSection.widgets[TMWidget].criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMSection.widgets[TMWidget].criteria.interfaces[CInterface]</i>	<object>	One CInterface object.	Optional
<i>TMSection.widgets[TMWidget].criteria.interfaces[CInterface].ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMSection.widgets[TMWidget].criteria.interfaces[CInterface].name</i>	<string>	Interface name.	Optional
<i>TMSection.widgets[TMWidget].criteria.interfaces[CInterface].ifindex</i>	<number>	Interface index.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_groups[CHostGroup]</i>	<object>	One CHostGroup object.	Optional

<i>TMSection.widgets[TMWidget].criteria.host_groups[CHostGroup].name</i>	<string>	Host group name.	Optional
<i>TMSection.widgets[TMWidget].criteria.host_groups[CHostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscps[CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscps[CDSCP].name</i>	<string>	DSCP name.	Optional
<i>TMSection.widgets[TMWidget].criteria.dscps[CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>TMSection.widgets[TMWidget].criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMSection.widgets[TMWidget].criteria.applications[CAApplication]</i>	<object>	One CApplication object.	Optional
<i>TMSection.widgets[TMWidget].criteria.applications[CAApplication].code</i>	<string>	Application code.	Optional
<i>TMSection.widgets[TMWidget].criteria.applications[CAApplication].name</i>	<string>	Application name.	Optional
<i>TMSection.widgets[TMWidget].criteria.applications[CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMSection.widgets[TMWidget].title</i>	<string>	Widget title.	
<i>TMSection.widgets[TMWidget].attributes</i>	<object>	Widget common attributes.	Optional
<i>TMSection.widgets[TMWidget].attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMSection.widgets[TMWidget].attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMSection.widgets[TMWidget].attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMSection.widgets[TMWidget].attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMSection.widgets[TMWidget].attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMSection.widgets[TMWidget].attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMSection.widgets[TMWidget].attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMSection.widgets[TMWidget].attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMSection.widgets[TMWidget].attributes.width</i>	<number>	Widget width.	Optional
<i>TMSection.widgets[TMWidget].attributes.height</i>	<number>	Widget height.	Optional
<i>TMSection.widgets[TMWidget].attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMSection.widgets[TMWidget].attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMSection.widgets[TMWidget].attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMSection.widgets[TMWidget].attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMSection.widgets[TMWidget].attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMSection.widgets[TMWidget].attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>TMSection.widgets[TMWidget].attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>TMSection.widgets[TMWidget].attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>TMSection.widgets[TMWidget].attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMSection.widgets[TMWidget].attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMSection.widgets[TMWidget].attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL

<i>TMSection.widgets[TMWidget].attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMSection.widgets[TMWidget].user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMSection.widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMSection.widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMSection.widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMSection.widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMSection.widgets[TMWidget].user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMSection.widgets[TMWidget].user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMSection.widgets[TMWidget].user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMSection.widgets[TMWidget].user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMSection.widgets[TMWidget].timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional
<i>TMSection.section_id</i>	<i>&lt;number&gt;</i>	Section ID.	
<i>TMSection.layout</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Internal section layout.	Optional
<i>TMSection.layout[TMFlowLine]</i>	<i>&lt;object&gt;</i>	One horizontal line of widgets.	Optional
<i>TMSection.layout[TMFlowLine].flow_items</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of line items.	Optional
<i>TMSection.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<i>&lt;object&gt;</i>	Object representing one layout item.	Optional
<i>TMSection.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<i>&lt;number&gt;</i>	Widget ID.	Optional
<i>TMSection.layout[TMFlowLine].attributes</i>	<i>&lt;object&gt;</i>	List of line attributes.	Optional
<i>TMSection.layout[TMFlowLine].attributes.wrappable</i>	<i>&lt;string&gt;</i>	Flag allowing wrapping.	Optional
<i>TMSection.layout[TMFlowLine].attributes.full_width</i>	<i>&lt;string&gt;</i>	Flag representing width of the layout line.	Optional

TMSection.layout[TMFlowLine].attributes.  
item\_spacing

<string>

Item spacing between widgets.

Optional

## Reporting: Get template sections

Get a list of sections in the template.

GET https://{device}/api/profiler/1.2/reporting/templates/{template\_id}/sections

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "widgets": [
      {
        "config": {
          "datasource": string,
          "visualization": string,
          "widget_type": string
        },
        "widget_id": number,
        "criteria": {
          "ports": [
            {
              "port": number,
              "protocol": number,
              "name": string
            }
          ],
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,
                "name": string
              },
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              },
              "dscp": {
                "name": string,
                "code_point": number
              }
            }
          ],
          "services": [
            {
              "name": string,
              "service_id": number
            }
          ],
          "port_groups": [
            {
              "name": string,
              "group_id": number
            }
          ],
          "comparison_time_frame": {
            "data_resolution": string,
            "refresh_interval": string,
            "type": string
          },
          "host_group_pairs": [
            {
              "server": {
                "name": string,
                "group_id": number
              },
              "client": {
                "name": string,
                "group_id": number
              }
            }
          ]
        }
      }
    ]
  }
]
```



```
}
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
```



```
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
```

```
{
  "server": {
    "mac": string,
    "ipaddr": string,
    "name": string
  },
  "client": {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
},
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge thickness": string.

```

```

    "display_host_group_type": string,
    "extend_to_zero": string,
    "collapsible": string,
    "high_threshold": string,
    "n_items": number,
    "colspan": number,
    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
}
]
]

```

Example:

```
[ ]
```

Property Name	Type	Description	Notes
<i>TMSections</i>	<array of <object>>	List of TMSection objects.	
<i>TMSections</i> [TMSection]	<object>	One TMSection object.	Optional
<i>TMSections</i> [TMSection].widgets	<array of <object>>	List of widgets that belong to the section.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget]	<object>	One TMWidget object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].config	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMSections</i> [TMSection].widgets[TMWidget].config.datasources	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMSections</i> [TMSection].widgets[TMWidget].config.visualization	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>TMSections</i> [TMSection].widgets[TMWidget].config.widget_type	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMSections</i> [TMSection].widgets[TMWidget].widget_id	<number>	Internal widget ID within a dashboard.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria	<object>	Query criteria for the widget.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.ports	<array of <object>>	Watched ports.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.ports[CProtoPort]	<object>	One CProtoPort object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.ports[CProtoPort].port	<number>	Port specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.ports[CProtoPort].protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.ports[CProtoPort].name	<string>	Protocol + port combination name.	Optional

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ]	<object>	One <i>CDSCPAppPort</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].port	<object>	Port specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].port.port	<number>	Port specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].port.protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].port.name	<string>	Protocol + port combination name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].app	<object>	Application specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].app.code	<string>	Application code.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].app.name	<string>	Application name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].dscp	<object>	DSCP specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].dscp.name	<string>	DSCP name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_app_ports[ <i>CDSCPAppPort</i> ].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.services	<array of <object>>	Watched services.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.services[ <i>CService</i> ]	<object>	One <i>CService</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.services[ <i>CService</i> ].name	<string>	Service name.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.services[ <i>CService</i> ].service_id	<number>	Service ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.port_groups	<array of <object>>	Watched port groups.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.port_groups[ <i>CPortGroup</i> ]	<object>	One <i>CPortGroup</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.port_groups[ <i>CPortGroup</i> ].name	<string>	Name of the port group.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.port_groups[ <i>CPortGroup</i> ].group_id	<number>	ID of the port group.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.comparison_time_frame	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.comparison_time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.comparison_time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.comparison_time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pairs	<array of <object>>	Watched group pairs.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pairs [ <i>CHostGroupPair</i> ]	<object>	One <i>CHostGroupPair</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pairs [ <i>CHostGroupPair</i> ].server	<object>	Server host group specification.	

<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.name	<string>	Host group name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].server.group_id	<number>	Host group ID.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client	<object>	Client host group specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.name	<string>	Host group name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pairs [CHostGroupPair].client.group_id	<number>	Host group ID.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.wan_group	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.traffic_expression	<string>	Traffic expression.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.split_direction	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.include_successes	<string>	Include successful requests in active directory report.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.include_non_optimized_sites	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.columns	<array of <number>>	List of column ID.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.columns[item]	<number>	Column ID.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer]	<object>	One CApplicationServer object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app	<object>	Application specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.code	<string>	Application code.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.name	<string>	Application name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server	<object>	Server specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_servers [CApplicationServer].server.name	<string>	Host name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.devices	<array of <object>>	Watched devices.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.devices[CDevice]	<object>	One CDevice object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.devices[CDevice].ipaddr	<string>	Device IP address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.devices[CDevice].name	<string>	Device name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_ports	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.application_ports [CApplicationPort]	<object>	One CApplicationPort object.	Optional

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].port	<object>	Port specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].port.port	<number>	Port specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].app	<object>	Application specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.application_ports [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.include_failures	<string>	Include failed requests in active directory report.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].port	<object>	Port specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].server	<object>	Server host specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].client	<object>	Client host specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_ports [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].interface	<object>	Interface specification.	

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscp_interfaces [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.time_frame	<object>	Widget time frame specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.time_frame.data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.time_frame.refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.time_frame.type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service	<object>	Watched service.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service.name	<string>	Service name.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service.service_id	<number>	Service ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.severity	<number>	Minimum severity filter for an event report.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.role	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.event_policies	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.event_policies[item]	<number>	Event policy ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_locations	<array of <object>>	Watched service locations.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_locations [CServiceLocation]	<object>	One CServiceLocation object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_locations [CServiceLocation].name	<string>	Service location name.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_locations [CServiceLocation].location_id	<string>	Service location ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.case_insensitive	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_location	<object>	Watched service location.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_location.name	<string>	Service location name.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.service_location.location_id	<string>	Service location ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.include_backend_segments	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_type	<string>	Host group type used.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_app_ports	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pair_app_ports [CHostPairAppPort]	<object>	One CHostPairAppPort object.	Optional



<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port	<object>	Port specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.port	<number>	Port specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app	<object>	Application specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.code	<string>	Application code.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.name	<string>	Application name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server	<object>	Server host specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].server.name	<string>	Host name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client	<object>	Client host specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_pair_app_ports [CHostPairAppPort].client.name	<string>	Host name.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.users	<array of <object>>	Watched users.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.users[CUser]	<object>	One CUser object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.users[CUser].name	<string>	Active Directory user name.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.sort_desc	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.sort_column	<number>	Sorting column ID.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMSections</i> [TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional



<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments	<array of <object>>	Watched network segments.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].src	<object>	Segment source.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.network_segments [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.hosts	<array of <object>>	Watched hosts.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.hosts [CHost]	<object>	One CHost object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.hosts [CHost].mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.hosts [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.hosts [CHost].name	<string>	Host name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs	<array of <object>>	Watched host pairs.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair].server	<object>	Specification of the server host.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair].server.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair].server.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair].server.name	<string>	Host name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs [CHostPair].client	<object>	Specification of the client host.	

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs[ <i>CHostPair</i> ].client.mac	<string>	Host MAC address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs[ <i>CHostPair</i> ].client.ipaddr	<string>	Host IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_pairs[ <i>CHostPair</i> ].client.name	<string>	Host name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.protocols	<array of <object>>	Watched protocols.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.protocols[ <i>CProtocol</i> ]	<object>	Object representing Protocol information.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.protocols[ <i>CProtocol</i> ].id	<number>	ID of the Protocol.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.protocols[ <i>CProtocol</i> ].name	<string>	Name of the Protocol.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.centricity	<string>	Centricity used to run the report.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.limit	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.interfaces	<array of <object>>	Watched interfaces.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.interfaces[ <i>CInterface</i> ]	<object>	One <i>CInterface</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.interfaces[ <i>CInterface</i> ].ipaddr	<string>	Interface IP address.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.interfaces[ <i>CInterface</i> ].name	<string>	Interface name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.interfaces[ <i>CInterface</i> ].ifindex	<number>	Interface index.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_groups	<array of <object>>	Watched host groups.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_groups[ <i>CHostGroup</i> ]	<object>	One <i>CHostGroup</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_groups[ <i>CHostGroup</i> ].name	<string>	Host group name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.host_groups[ <i>CHostGroup</i> ].group_id	<number>	Host group ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscps	<array of <object>>	Watched DSCPs.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscps[ <i>CDSCP</i> ]	<object>	One <i>CDSCP</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscps[ <i>CDSCP</i> ].name	<string>	DSCP name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.dscps[ <i>CDSCP</i> ].code_point	<number>	DSCP code point.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.applications	<array of <object>>	Watched applications.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.applications[ <i>CApplication</i> ]	<object>	One <i>CApplication</i> object.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.applications[ <i>CApplication</i> ].code	<string>	Application code.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.applications[ <i>CApplication</i> ].name	<string>	Application name.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].criteria.applications[ <i>CApplication</i> ].tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].title	<string>	Widget title.	
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes	<object>	Widget common attributes.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.width	<number>	Widget width.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.height	<number>	Widget height.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes	<object>	User-specific attributes.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.pan_zoomable	<string>	Flag making the graph interactive.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.line_scale	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.format_bytes	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.show_images	<string>	Flag showing images in a connection graph.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.open_nodes	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.open_nodes[item]	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.line_style	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.layout	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.width	<number>	Widget width.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.height	<number>	Widget height.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.percent_of_total	<string>	Flag including the 'total' item in a pie chart.	Optional

<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.edge_thickness	<string>	Widget edge thickness.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.display_host_group_type	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.extend_to_zero	<string>	Flag: extending the Y-axis to zero.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.collapsible	<string>	Flag indicating if the widget is collapsible.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.high_threshold	<string>	High threshold on the chart (in bytes).	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.n_items	<number>	Maximum number of items shown.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.colspan	<number>	How many columns the widget occupies in layout.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.low_threshold	<string>	Low threshold on the chart (in bytes).	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.moveable_nodes	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.orientation	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].user_attributes.modal_links	<number>	Flag adding modal links on a widget.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].widgets[ <i>TMWidget</i> ].timestamp	<string>	Widget time stamp specification.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].section_id	<number>	Section ID.	
<i>TMSections</i> [ <i>TMSection</i> ].layout	<array of <object>>	Internal section layout.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ]	<object>	One horizontal line of widgets.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].flow_items	<array of <object>>	List of line items.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].flow_items[ <i>TMFlowItem</i> ]	<object>	Object representing one layout item.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].flow_items[ <i>TMFlowItem</i> ].id	<number>	Widget ID.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].attributes	<object>	List of line attributes.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].attributes.wrappable	<string>	Flag allowing wrapping.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].attributes.full_width	<string>	Flag representing width of the layout line.	Optional
<i>TMSections</i> [ <i>TMSection</i> ].layout[ <i>TMFlowLine</i> ].attributes.item_spacing	<string>	Item spacing between widgets.	Optional

## Reporting: Get template

Get a template.

```
GET https://{device}/api/profiler/1.2/reporting/templates/{template_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "traffic_expression": string,
  "id": number,
  "scheduled": string,
  "sharing": {
    "users": [
      number
    ]
  },
  "layout": [
    {
      "flow_items": [
```

```
{
  "id": number
},
],
"attributes": {
  "wrappable": string,
  "full_width": string,
  "item_spacing": string
}
},
],
"description": string,
"user_id": number,
"shared": string,
"live": string,
"last_added_section_id": number,
"name": string,
"last_added_widget_id": number,
"version": string,
"disabled": string,
"timestamp": string,
"sections": [
  {
    "widgets": [
      {
        "config": {
          "datasource": string,
          "visualization": string,
          "widget_type": string
        },
        "widget_id": number,
        "criteria": {
          "ports": [
            {
              "port": number,
              "protocol": number,
              "name": string
            }
          ],
          "dscp_app_ports": [
            {
              "port": {
                "port": number,
                "protocol": number,
                "name": string
              },
              "app": {
                "code": string,
                "name": string,
                "tunneled": string
              },
              "dscp": {
                "name": string,
                "code_point": number
              }
            }
          ],
          "services": [
            {
              "name": string,
              "service_id": number
            }
          ],
          "port_groups": [
            {
              "name": string,
              "group_id": number
            }
          ],
          "comparison_time_frame": {
            "data_resolution": string,
            "refresh_interval": string,
            "type": string
          },
          "host_group_pairs": [
            {
              "server": {
                "name": string,
                "group_id": number
              },
              "client": {
                "name": string,
                "group_id": number
              }
            }
          ],
          "wan_group": string,
          "traffic_expression": string
        }
      }
    ]
  }
]
```

```
    "traffic_expression": string,
    "split_direction": string,
    "include_successes": string,
    "include_non_optimized_sites": string,
    "columns": [
      number
    ],
    "application_servers": [
      {
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        },
        "server": {
          "mac": string,
          "ipaddr": string,
          "name": string
        }
      }
    ],
    "devices": [
      {
        "ipaddr": string,
        "name": string
      }
    ],
    "application_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        }
      }
    ],
    "include_failures": string,
    "host_pair_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "server": {
          "mac": string,
          "ipaddr": string,
          "name": string
        },
        "client": {
          "mac": string,
          "ipaddr": string,
          "name": string
        }
      }
    ],
    "dscp_interfaces": [
      {
        "interface": {
          "ipaddr": string,
          "name": string,
          "ifindex": number
        },
        "dscp": {
          "name": string,
          "code_point": number
        }
      }
    ],
    "time_frame": {
      "data_resolution": string,
      "refresh_interval": string,
      "type": string
    },
    "service": {
      "name": string,
      "service_id": number
    },
    "severity": number,
    "role": string,
    "event_policies": [
      number
    ]
  }
}
```

```

"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    }
  }
],
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,

```

```
    "name": string
  },
  "client": {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
},
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
```



```

    "high_threshold": string,
    "n_items": number,
    "colspan": number,
    "low_threshold": string,
    "moveable_nodes": string,
    "orientation": string,
    "modal_links": number
  },
  "timestamp": string
}
],
"section_id": number,
"layout": [
  {
    "flow_items": [
      {
        "id": number
      }
    ],
    "attributes": {
      "wrappable": string,
      "full_width": string,
      "item_spacing": string
    }
  }
]
},
"img": {
  "thumbnail": {
    "src": string
  },
  "full": {
    "src": string
  }
}
}
}

```

Example:

```

{
  "layout": [
    {
      "flow_items": [
        {
          "id": 1
        }
      ]
    }
  ],
  "name": "VOIP - Call Quality and Usage",
  "user_id": 1,
  "timestamp": "1383141976.674345",
  "live": true,
  "last_added_widget_id": 6,
  "traffic_expression": "app VoIP-RTP",
  "version": "1.1",
  "shared": "Private",
  "sections": [
    {
      "widgets": [
        {
          "title": "VoIP-RTP: Applications",
          "timestamp": "1383141976.674383",
          "criteria": {
            "sort_column": 33,
            "traffic_expression": "",
            "sort_desc": true,
            "centricity": "host",
            "limit": 100,
            "columns": [
              17,
              33,
              34,
              757,
              766,
              781,
              803
            ],
            "time_frame": {
              "data_resolution": "15mins",
              "type": "last_hour",
              "refresh_interval": "15mins"
            }
          },
          "attributes": {
            "format_bytes": "UI_PREF",
            "colspan": 2,
            "rowspan": 2
          }
        }
      ]
    }
  ]
}

```

```
"n_items": 20
},
"config": {
  "widget_type": "APPS",
  "visualization": "TABLE",
  "datasource": "TRAFFIC"
},
"widget_id": 1
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674428",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      803
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_scale": "LINEAR",
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 2
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674459",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      781
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 1,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 3
},
{
  "title": "VoIP-RTP: Traffic Quality",
  "timestamp": "1383141976.674497",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      766
    ],
    "time_frame": {
      "data_resolution": "min",
      "type": "last_hour",
      "refresh_interval": "min"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false
```

```
    extend_to_zero : raise,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 4
},
{
  "title": "VoIP-RTP: Traffic Volume",
  "timestamp": "1383141976.674527",
  "criteria": {
    "traffic_expression": "",
    "sort_desc": true,
    "centricity": "host",
    "columns": [
      33
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_day",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "extend_to_zero": false,
    "line_style": "STACKED"
  },
  "config": {
    "widget_type": "TRAFFIC_OVERALL",
    "visualization": "LINE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 5
},
{
  "title": "Host Group Pairs",
  "timestamp": "1383141976.674566",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "host_group_type": "ByLocation",
    "sort_desc": true,
    "centricity": "host",
    "limit": 100,
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "show_images": true,
    "layout": "HORIZONTAL_TREE",
    "colspan": 2,
    "moveable_nodes": true,
    "height": 400,
    "edge_thickness": true,
    "pan_zoomable": true,
    "n_items": 10
  },
  "config": {
    "widget_type": "HOST_GROUP_PAIRS",
    "visualization": "CONN_GRAPH",
    "datasource": "TRAFFIC"
  },
  "widget_id": 6
}
],
"layout": [
  {
    "flow_items": [
      {
        "id": 1
      }
    ]
  },
  {
    "flow_items": [
      {
        "id": 2
      }
    ]
  }
]
```

```

    "id": 3
  }
],
{
  "flow_items": [
    {
      "id": 4
    }
  ]
},
{
  "flow_items": [
    {
      "id": 5
    }
  ]
},
{
  "flow_items": [
    {
      "id": 6
    }
  ]
}
],
"section_id": 1
}
],
"id": 5217,
"description": ""
}

```

Property Name	Type	Description	Notes
<i>ReportTemplateSpec</i>	<object>	Reporting template specification object.	
<i>ReportTemplateSpec.traffic_expression</i>	<string>	Traffic expression applied to all widgets within the template.	Optional
<i>ReportTemplateSpec.id</i>	<number>	ID of the report template.	Optional
<i>ReportTemplateSpec.scheduled</i>	<string>	Flag indicating that the template is scheduled.	Optional
<i>ReportTemplateSpec.sharing</i>	<object>	List of the users the template is shared with (see <i>ReportTemplateSharing</i> ).	Optional
<i>ReportTemplateSpec.sharing.users</i>	<array of <number>>	List of the users a template is shared with.	Optional
<i>ReportTemplateSpec.sharing.users[item]</i>	<number>	User ID.	Optional
<i>ReportTemplateSpec.layout</i>	<array of <object>>	Layout information.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine]</i>	<object>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items</i>	<array of <object>>	List of line items.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem]</i>	<object>	Object representing one layout item.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].flow_items[TMFlowItem].id</i>	<number>	Widget ID.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes</i>	<object>	List of line attributes.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.wrappable</i>	<string>	Flag allowing wrapping.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.full_width</i>	<string>	Flag representing width of the layout line.	Optional
<i>ReportTemplateSpec.layout[TMFlowLine].attributes.item_spacing</i>	<string>	Item spacing between widgets.	Optional
<i>ReportTemplateSpec.description</i>	<string>	Human-readable description of the template.	Optional
<i>ReportTemplateSpec.user_id</i>	<number>	User ID of the template owner.	Optional
<i>ReportTemplateSpec.shared</i>	<string>	Flag indicating that the template is shared with other users.	Optional; Values: Private, Public, Users
<i>ReportTemplateSpec.live</i>	<string>	Flag indicating that the template is a dashboard.	
<i>ReportTemplateSpec.last_added_section_id</i>	<number>	ID of the last layout section added to the template.	Optional
<i>ReportTemplateSpec.name</i>	<string>	Human-readable name of the template.	
<i>ReportTemplateSpec.last_added_widget_id</i>	<number>	ID of the last widget added to the template.	Optional
<i>ReportTemplateSpec.version</i>	<string>	Version of the specification.	Optional
<i>ReportTemplateSpec.disabled</i>	<string>	Flag indicating that the template is disabled.	Optional

<i>ReportTemplateSpec.timestamp</i>	<string>	Report time stamp (unix time).	Optional
<i>ReportTemplateSpec.sections</i>	<array of <object>>	List of layout sections.	Optional
<i>ReportTemplateSpec.sections[TMSection]</i>	<object>	One TMSection object.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets</i>	<array of <object>>	List of widgets that belong to the section.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget]</i>	<object>	One TMWidget object.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].config.datasources</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRIT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria</i>	<object>	Query criteria for the widget.	
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria.ports [CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria.ports [CProtoPort].port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria.ports [CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria.ports [CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort]</i>	<object>	One CDSCAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].port. protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection]. widgets[TMWidget].criteria. dscp_app_ports[CDSCAppPort].app.name</i>	<string>	Application name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp</i>	<object>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_app_ports[CDSCAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService].name</i>	<string>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.port_groups[CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.name</i>	<string>	Host group name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pairs[CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server</i>	<object>	Server specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_servers[CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.application_ports[CApplicationPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort]</i>	<object>	One CHostPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client</i>	<object>	Client host specification.	



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.mac</i>	<i>&lt;string&gt;</i>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.ipaddr</i>	<i>&lt;string&gt;</i>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_ports[CHostPairPort].client.name</i>	<i>&lt;string&gt;</i>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Watched combinations of DSCPs and interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface]</i>	<i>&lt;object&gt;</i>	One CDSCPInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface</i>	<i>&lt;object&gt;</i>	Interface specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ipaddr</i>	<i>&lt;string&gt;</i>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.name</i>	<i>&lt;string&gt;</i>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].interface.ifindex</i>	<i>&lt;number&gt;</i>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp</i>	<i>&lt;object&gt;</i>	DSCP specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.name</i>	<i>&lt;string&gt;</i>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscp_interfaces[CDSCPInterface].dscp.code_point</i>	<i>&lt;number&gt;</i>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame</i>	<i>&lt;object&gt;</i>	Widget time frame specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.data_resolution</i>	<i>&lt;string&gt;</i>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.refresh_interval</i>	<i>&lt;string&gt;</i>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.time_frame.type</i>	<i>&lt;string&gt;</i>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service</i>	<i>&lt;object&gt;</i>	Watched service.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.name</i>	<i>&lt;string&gt;</i>	Service name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service.service_id</i>	<i>&lt;number&gt;</i>	Service ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.severity</i>	<i>&lt;number&gt;</i>	Minimum severity filter for an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.role</i>	<i>&lt;string&gt;</i>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies</i>	<i>&lt;array of &lt;number&gt;&gt;</i>	List of event policies to include in an event report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.event_policies[item]</i>	<i>&lt;number&gt;</i>	Event policy ID.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_locations[CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location</i>	<object>	Watched service location.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.name</i>	<string>	Service location name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app</i>	<object>	Application specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pair_app_ports[CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser]</i>	<object>	One CUser object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.users [CUser].name</i>	<string>	Active Directory user name.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.sort_column</i>	<number>	Sorting column ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort]</i>	<object>	One CHostGroupPairPort object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port</i>	<object>	Port specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.port</i>	<number>	Port specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server</i>	<object>	Server host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].server.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client</i>	<object>	Client host group specification.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.name</i>	<string>	Host group name.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_group_pair_ports [CHostGroupPairPort].client.group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment]</i>	<object>	One CNetworkSegment object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src</i>	<object>	Segment source.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].src.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst</i>	<object>	Segment destination.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.network_segments[CNetworkSegment].dst.ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost]</i>	<object>	One CHost object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.hosts [CHost].name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair]</i>	<object>	One CHostPair object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server</i>	<object>	Specification of the server host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.mac</i>	<string>	Host MAC address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].server.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client</i>	<object>	Specification of the client host.	
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.mac</i>	<string>	Host MAC address.	Optional

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_pairs [CHostPair].client.name</i>	<string>	Host name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol]</i>	<object>	Object representing Protocol information.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].id</i>	<number>	ID of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.protocols [CProtocol].name</i>	<string>	Name of the Protocol.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface]</i>	<object>	One CInterface object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ipaddr</i>	<string>	Interface IP address.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].name</i>	<string>	Interface name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.interfaces [CInterface].ifindex</i>	<number>	Interface index.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup]</i>	<object>	One CHostGroup object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].name</i>	<string>	Host group name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.host_groups [CHostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].name</i>	<string>	DSCP name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.dscps [CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications [CAApplication]</i>	<object>	One CAApplication object.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications [CAApplication].code</i>	<string>	Application code.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications [CAApplication].name</i>	<string>	Application name.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].criteria.applications [CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].title</i>	<string>	Widget title.	

<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes</i>	<object>	Widget common attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.width</i>	<number>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.height</i>	<number>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.collapsible</i>	<string>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.high_threshold</i>	<string>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.n_items</i>	<number>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.colspan</i>	<number>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.low_threshold</i>	<string>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.moveable_nodes</i>	<string>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.orientation</i>	<string>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].attributes.modal_links</i>	<number>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes</i>	<object>	User-specific attributes.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG



<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBITS, KBYTES, MBITS, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.open_nodes[item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>ReportTemplateSpec.sections[TMSection].widgets[TMWidget].timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional
<i>ReportTemplateSpec.sections[TMSection].section_id</i>	<i>&lt;number&gt;</i>	Section ID.	
<i>ReportTemplateSpec.sections[TMSection].layout</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Internal section layout.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine]</i>	<i>&lt;object&gt;</i>	One horizontal line of widgets.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of line items.	Optional
<i>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items [TMFlowItem]</i>	<i>&lt;object&gt;</i>	Object representing one layout item.	Optional

<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].flow_items[TMFlowItem].id</code>	<code>&lt;number&gt;</code>	Widget ID.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes</code>	<code>&lt;object&gt;</code>	List of line attributes.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.wrappable</code>	<code>&lt;string&gt;</code>	Flag allowing wrapping.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.full_width</code>	<code>&lt;string&gt;</code>	Flag representing width of the layout line.	Optional
<code>ReportTemplateSpec.sections[TMSection].layout[TMFlowLine].attributes.item_spacing</code>	<code>&lt;string&gt;</code>	Item spacing between widgets.	Optional
<code>ReportTemplateSpec.img</code>	<code>&lt;object&gt;</code>	Images associated with the template.	Optional
<code>ReportTemplateSpec.img.thumbnail</code>	<code>&lt;object&gt;</code>	A thumbnail-size image for the report template.	Optional
<code>ReportTemplateSpec.img.thumbnail.src</code>	<code>&lt;string&gt;</code>	Relative URL of an image.	
<code>ReportTemplateSpec.img.full</code>	<code>&lt;object&gt;</code>	A full-size image for the report template.	Optional
<code>ReportTemplateSpec.img.full.src</code>	<code>&lt;string&gt;</code>	Relative URL of an image.	

## Reporting: Get report view (PDF, CSV)

Get GUI view of a report (PDF, CSV).

```
GET https://{device}/api/profiler/1.2/reporting/reports/{report_id}/view
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Reporting: List roles

Get a list of roles that this version of the API supports.

```
GET https://{device}/api/profiler/1.2/reporting/roles
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": string,
    "name": string
  }
]
```

Example:

```
[
  {
    "id": "cli",
    "name": "client"
  },
  {
    "id": "srv",
    "name": "server"
  }
]
```

Property Name	Type	Description	Notes
<i>Roles</i>	<code>&lt;array of &lt;object&gt;&gt;</code>	List of roles.	



Roles[Role]	<object>	Object representing a roles.	Optional
Roles[Role].id	<string>	ID of a role. To be used in the API.	
Roles[Role].name	<string>	Human-readable name of a role.	

## Reporting: Copy widget

Copy a widget from one template to another.

POST [https://{device}/api/profiler/1.2/reporting/templates/widgets/copy?dest\\_template={number}&src\\_template={number}](https://{device}/api/profiler/1.2/reporting/templates/widgets/copy?dest_template={number}&src_template={number})

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
widget	<number>	ID of the widget being copied.	
dest_template	<number>	Destination template ID.	
src_template	<number>	Source template ID.	

### Request Body

Do not provide a request body.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "config": {
    "datasource": string,
    "visualization": string,
    "widget_type": string
  },
  "widget_id": number,
  "criteria": {
    "ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "dscp_app_ports": [
      {
        "port": {
          "port": number,
          "protocol": number,
          "name": string
        },
        "app": {
          "code": string,
          "name": string,
          "tunneled": string
        },
        "dscp": {
          "name": string,
          "code_point": number
        }
      }
    ],
    "services": [
      {
        "name": string,
        "service_id": number
      }
    ],
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "comparison_time_frame": {
      "data_resolution": string,
      "start_time": string
    }
  }
}
```

```
"reresh_interval": string,
"type": string
},
"host_group_pairs": [
  {
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"wan_group": string,
"traffic_expression": string,
"split_direction": string,
"include_successes": string,
"include_non_optimized_sites": string,
"columns": [
  number
],
"application_servers": [
  {
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"devices": [
  {
    "ipaddr": string,
    "name": string
  }
],
"application_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    }
  }
],
"include_failures": string,
"host_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"dscp_interfaces": [
  {
    "interface": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dscp": {
      "name": string,
      "code_point": number
    }
  }
]
```

```
},
},
],
"time_frame": {
  "data_resolution": string,
  "refresh_interval": string,
  "type": string
},
"service": {
  "name": string,
  "service_id": number
},
"severity": number,
"role": string,
"event_policies": [
  number
],
"service_locations": [
  {
    "name": string,
    "location_id": string
  }
],
"case_insensitive": string,
"service_location": {
  "name": string,
  "location_id": string
},
"include_backend_segments": string,
"host_group_type": string,
"host_pair_app_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "app": {
      "code": string,
      "name": string,
      "tunneled": string
    },
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"users": [
  {
    "name": string
  }
],
"sort_desc": string,
"sort_column": number,
"host_group_pair_ports": [
  {
    "port": {
      "port": number,
      "protocol": number,
      "name": string
    },
    "server": {
      "name": string,
      "group_id": number
    },
    "client": {
      "name": string,
      "group_id": number
    }
  }
],
"network_segments": [
  {
    "src": {
      "ipaddr": string,
      "name": string,
      "ifindex": number
    },
    "dst": {
      "ipaddr": string,
      "name": string.
    }
  }
]
```

```
    "ifindex": number
  }
},
"hosts": [
  {
    "mac": string,
    "ipaddr": string,
    "name": string
  }
],
"host_pairs": [
  {
    "server": {
      "mac": string,
      "ipaddr": string,
      "name": string
    },
    "client": {
      "mac": string,
      "ipaddr": string,
      "name": string
    }
  }
],
"protocols": [
  {
    "id": number,
    "name": string
  }
],
"centricity": string,
"limit": number,
"interfaces": [
  {
    "ipaddr": string,
    "name": string,
    "ifindex": number
  }
],
"host_groups": [
  {
    "name": string,
    "group_id": number
  }
],
"dscps": [
  {
    "name": string,
    "code_point": number
  }
],
"applications": [
  {
    "code": string,
    "name": string,
    "tunneled": string
  }
]
},
"title": string,
"attributes": {
  "pan_zoomable": string,
  "line_scale": string,
  "format_bytes": string,
  "show_images": string,
  "open_nodes": [
    string
  ],
  "line_style": string,
  "layout": string,
  "width": number,
  "height": number,
  "percent_of_total": string,
  "edge_thickness": string,
  "display_host_group_type": string,
  "extend_to_zero": string,
  "collapsible": string,
  "high_threshold": string,
  "n_items": number,
  "colspan": number,
  "low_threshold": string,
  "moveable_nodes": string,
  "orientation": string,
  "modal_links": number
},
"user_attributes": {
```

```

"pan_zoomable": string,
"line_scale": string,
"format_bytes": string,
"show_images": string,
"open_nodes": [
  string
],
"line_style": string,
"layout": string,
"width": number,
"height": number,
"percent_of_total": string,
"edge_thickness": string,
"display_host_group_type": string,
"extend_to_zero": string,
"collapsible": string,
"high_threshold": string,
"n_items": number,
"colspan": number,
"low_threshold": string,
"moveable_nodes": string,
"orientation": string,
"modal_links": number
},
"timestamp": string
}

```

Example:

```

{
  "title": "VoIP-RTP: Applications",
  "timestamp": "1383141976.674383",
  "criteria": {
    "sort_column": 33,
    "traffic_expression": "",
    "centricity": "host",
    "limit": 100,
    "columns": [
      17,
      33,
      34,
      757,
      766,
      781,
      803
    ],
    "time_frame": {
      "data_resolution": "15mins",
      "type": "last_hour",
      "refresh_interval": "15mins"
    }
  },
  "attributes": {
    "format_bytes": "UI_PREF",
    "colspan": 2,
    "n_items": 20
  },
  "config": {
    "widget_type": "APPS",
    "visualization": "TABLE",
    "datasource": "TRAFFIC"
  },
  "widget_id": 1
}

```

Property Name	Type	Description	Notes
<i>TMWidget</i>	<object>	Widget specification.	
<i>TMWidget.config</i>	<object>	Widget configuration: data source type, widget type, and visualization type.	
<i>TMWidget.config.datasource</i>	<string>	Data source type.	Values: TRAFFIC, WAN, SERVICE, EVENTS, ACTIVE_DIRECTORY
<i>TMWidget.config.visualization</i>	<string>	Visualization type.	Values: TABLE, PIE, BAR, LINE, CONN_GRAPH, TREE_TABLE, INVISIBLE

<i>TMWidget.config.widget_type</i>	<string>	Widget type.	Values: TRAFFIC_OVERALL, TRAFFIC_OVERALL_CRTT, HOSTS, PEER_HOSTS, PEER_HOST_GROUPS, HOST_PAIRS_PORTS, HOST_PAIRS_APP_PORTS, HOST_PAIRS, HOST_GROUPS, HOST_GROUP_PAIRS, HOST_GROUP_PAIR_PORTS, APPS, APP_PORTS, SERVER_APPS, PORTS, PORT_GROUPS, PROTOCOLS, DEVICES, INTERFACES, NETWORK_SEGMENTS, DSCPS, DSCP_APP_PORTS, DSCP_IFACES, CURRENT_EVENTS, UNACKED_EVENTS, ACTIVE_DIRECTORY, SERVICE_HEALTH, LOCATION_HEALTH, SERVICE_HEALTH_MAP, LOCATION_HEALTH_MAP
<i>TMWidget.widget_id</i>	<number>	Internal widget ID within a dashboard.	Optional
<i>TMWidget.criteria</i>	<object>	Query criteria for the widget.	
<i>TMWidget.criteria.ports</i>	<array of <object>>	Watched ports.	Optional
<i>TMWidget.criteria.ports[CProtoPort]</i>	<object>	One CProtoPort object.	Optional
<i>TMWidget.criteria.ports[CProtoPort].port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.ports[CProtoPort].name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports</i>	<array of <object>>	Watched combinations of DSCPs, applications, and ports.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort]</i>	<object>	One CDSCPAppPort object.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp</i>	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_app_ports [CDSCPAppPort].dscp.code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.services</i>	<array of <object>>	Watched services.	Optional
<i>TMWidget.criteria.services[CService]</i>	<object>	One CService object.	Optional
<i>TMWidget.criteria.services[CService].name</i>	<string>	Service name.	
<i>TMWidget.criteria.services[CService].service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.port_groups</i>	<array of <object>>	Watched port groups.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup]</i>	<object>	One CPortGroup object.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].name</i>	<string>	Name of the port group.	Optional
<i>TMWidget.criteria.port_groups [CPortGroup].group_id</i>	<number>	ID of the port group.	Optional
<i>TMWidget.criteria.comparison_time_frame</i>	<object>	Alternative time frame specification to be used in a comparison widget.	Optional
<i>TMWidget.criteria.comparison_time_frame.data_resolution</i>	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.comparison_time_frame.refresh_interval</i>	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month

<i>TMWidget.criteria.comparison_time_frame.type</i>	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.host_group_pairs</i>	<array of <object>>	Watched group pairs.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair]</i>	<object>	One CHostGroupPair object.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server</i>	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].server.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client</i>	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pairs [CHostGroupPair].client.group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.wan_group</i>	<string>	WAN group used in WAN Optimization widgets. Can be one of '/WAN', '/WAN/Optimized', '/WAN/Non-optimized'.	Optional
<i>TMWidget.criteria.traffic_expression</i>	<string>	Traffic expression.	Optional
<i>TMWidget.criteria.split_direction</i>	<string>	Split inbound/outbound or received/transmitted data.	Optional
<i>TMWidget.criteria.include_successes</i>	<string>	Include successful requests in active directory report.	Optional
<i>TMWidget.criteria.include_non_optimized_sites</i>	<string>	Flag indicating whether to include WAN non optimized sites.	Optional
<i>TMWidget.criteria.columns</i>	<array of <number>>	List of column ID.	Optional
<i>TMWidget.criteria.columns[item]</i>	<number>	Column ID.	Optional
<i>TMWidget.criteria.application_servers</i>	<array of <object>>	Watched combinations of applications and servers.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer]</i>	<object>	One CApplicationServer object.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server</i>	<object>	Server specification.	
<i>TMWidget.criteria.application_servers [CApplicationServer].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.application_servers [CApplicationServer].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.devices</i>	<array of <object>>	Watched devices.	Optional
<i>TMWidget.criteria.devices[CDevice]</i>	<object>	One CDevice object.	Optional
<i>TMWidget.criteria.devices[CDevice].ipaddr</i>	<string>	Device IP address.	Optional
<i>TMWidget.criteria.devices[CDevice].name</i>	<string>	Device name.	Optional
<i>TMWidget.criteria.application_ports</i>	<array of <object>>	Watched combinations of applications and ports.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort]</i>	<object>	One CApplicationPort object.	Optional
<i>TMWidget.criteria.application_ports [CApplicationPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.application_ports [CApplicationPort].port.port</i>	<number>	Port specification.	Optional

<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app	<object>	Application specification.	
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.code	<string>	Application code.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.name	<string>	Application name.	Optional
<i>TMWidget.criteria.application_ports</i> [CApplicationPort].app.tunneled	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.include_failures</i>	<string>	Include failed requests in active directory report.	Optional
<i>TMWidget.criteria.host_pair_ports</i>	<array of <object>>	Watched combinations of host pairs and ports.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort]	<object>	One CHostPairPort object.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_ports</i> [CHostPairPort].client.name	<string>	Host name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i>	<array of <object>>	Watched combinations of DSCPs and interfaces.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface]	<object>	One CDSCPInterface object.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface	<object>	Interface specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].interface.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp	<object>	DSCP specification.	
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.name	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dscp_interfaces</i> [CDSCPInterface].dscp.code_point	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.time_frame</i>	<object>	Widget time frame specification.	Optional
<i>TMWidget.criteria.time_frame</i> . data_resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> . refresh_interval	<string>	Report refresh interval. It can be one of: min, 15mins, hour, 6hours, day, week, month.	Optional; Values: min, 15mins, hour, 6hours, day, week, month
<i>TMWidget.criteria.time_frame</i> .type	<string>	Type of time frame. Can be one of: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month.	Optional; Values: last_min, last_15mins, last_hour, last_6hours, last_12hours, last_day, last_week, last_month, previous_hour, previous_day, previous_week, previous_month
<i>TMWidget.criteria.service</i>	<object>	Watched service.	Optional



<i>TMWidget.criteria.service.name</i>	<string>	Service name.	
<i>TMWidget.criteria.service.service_id</i>	<number>	Service ID.	Optional
<i>TMWidget.criteria.severity</i>	<number>	Minimum severity filter for an event report.	Optional
<i>TMWidget.criteria.role</i>	<string>	Which host roles to include in a report ('CLIENT_SERVER', 'CLIENT', 'SERVER').	Optional; Values: CLIENT_SERVER, CLIENT, SERVER
<i>TMWidget.criteria.event_policies</i>	<array of <number>>	List of event policies to include in an event report.	Optional
<i>TMWidget.criteria.event_policies[item]</i>	<number>	Event policy ID.	Optional
<i>TMWidget.criteria.service_locations</i>	<array of <object>>	Watched service locations.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation]</i>	<object>	One CServiceLocation object.	Optional
<i>TMWidget.criteria.service_locations [CServiceLocation].name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_locations [CServiceLocation].location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.case_insensitive</i>	<string>	Case-insensitive usernames in an identity report.	Optional
<i>TMWidget.criteria.service_location</i>	<object>	Watched service location.	Optional
<i>TMWidget.criteria.service_location.name</i>	<string>	Service location name.	
<i>TMWidget.criteria.service_location.location_id</i>	<string>	Service location ID.	Optional
<i>TMWidget.criteria.include_backend_segments</i>	<string>	Flag indicating whether to include back-end segments.	Optional
<i>TMWidget.criteria.host_group_type</i>	<string>	Host group type used.	Optional
<i>TMWidget.criteria.host_pair_app_ports</i>	<array of <object>>	Watched combinations of host pairs, applications, and ports.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort]</i>	<object>	One CHostPairAppPort object.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port</i>	<object>	Port specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.port</i>	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.protocol</i>	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].port.name</i>	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app</i>	<object>	Application specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].app.tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server</i>	<object>	Server host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].server.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client</i>	<object>	Client host specification.	
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.mac</i>	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.ipaddr</i>	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pair_app_ports [CHostPairAppPort].client.name</i>	<string>	Host name.	Optional
<i>TMWidget.criteria.users</i>	<array of <object>>	Watched users.	Optional
<i>TMWidget.criteria.users[CUser]</i>	<object>	One CUser object.	Optional
<i>TMWidget.criteria.users[CUser].name</i>	<string>	Active Directory user name.	
<i>TMWidget.criteria.sort_desc</i>	<string>	Sorting direction (true for descending, false for ascending).	Optional
<i>TMWidget.criteria.sort_column</i>	<number>	Sorting column ID.	Optional

<i>TMWidget.criteria.host_group_pair_ports</i>	<array of <object>>	Watched combinations of host groups pairs and ports.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort]	<object>	One CHostGroupPairPort object.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port	<object>	Port specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.port	<number>	Port specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.protocol	<number>	Protocol specification.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].port.name	<string>	Protocol + port combination name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server	<object>	Server host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].server.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client	<object>	Client host group specification.	
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.name	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_group_pair_ports</i> [CHostGroupPairPort].client.group_id	<number>	Host group ID.	Optional
<i>TMWidget.criteria.network_segments</i>	<array of <object>>	Watched network segments.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment]	<object>	One CNetworkSegment object.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src	<object>	Segment source.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].src.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst	<object>	Segment destination.	
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ipaddr	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.name	<string>	Interface name.	Optional
<i>TMWidget.criteria.network_segments</i> [CNetworkSegment].dst.ifindex	<number>	Interface index.	Optional
<i>TMWidget.criteria.hosts</i>	<array of <object>>	Watched hosts.	Optional
<i>TMWidget.criteria.hosts</i> [CHost]	<object>	One CHost object.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.hosts</i> [CHost].name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i>	<array of <object>>	Watched host pairs.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]	<object>	One CHostPair object.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server	<object>	Specification of the server host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. server.name	<string>	Host name.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client	<object>	Specification of the client host.	
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.mac	<string>	Host MAC address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.ipaddr	<string>	Host IP address.	Optional
<i>TMWidget.criteria.host_pairs</i> [CHostPair]. client.name	<string>	Host name.	Optional

<i>TMWidget.criteria.protocols</i>	<array of <object>>	Watched protocols.	Optional
<i>TMWidget.criteria.protocols[CProtocol]</i>	<object>	Object representing Protocol information.	Optional
<i>TMWidget.criteria.protocols[CProtocol].id</i>	<number>	ID of the Protocol.	Optional
<i>TMWidget.criteria.protocols[CProtocol].name</i>	<string>	Name of the Protocol.	Optional
<i>TMWidget.criteria.centricity</i>	<string>	Centricity used to run the report.	Optional
<i>TMWidget.criteria.limit</i>	<number>	Maximum number of data rows in the report for the widget.	Optional
<i>TMWidget.criteria.interfaces</i>	<array of <object>>	Watched interfaces.	Optional
<i>TMWidget.criteria.interfaces[CInterface]</i>	<object>	One CInterface object.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ipaddr</i>	<string>	Interface IP address.	Optional
<i>TMWidget.criteria.interfaces[CInterface].name</i>	<string>	Interface name.	Optional
<i>TMWidget.criteria.interfaces[CInterface].ifindex</i>	<number>	Interface index.	Optional
<i>TMWidget.criteria.host_groups</i>	<array of <object>>	Watched host groups.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup]</i>	<object>	One CHostGroup object.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].name</i>	<string>	Host group name.	Optional
<i>TMWidget.criteria.host_groups [CHostGroup].group_id</i>	<number>	Host group ID.	Optional
<i>TMWidget.criteria.dsccps</i>	<array of <object>>	Watched DSCPs.	Optional
<i>TMWidget.criteria.dsccps[CDSCP]</i>	<object>	One CDSCP object.	Optional
<i>TMWidget.criteria.dsccps[CDSCP].name</i>	<string>	DSCP name.	Optional
<i>TMWidget.criteria.dsccps[CDSCP].code_point</i>	<number>	DSCP code point.	Optional
<i>TMWidget.criteria.applications</i>	<array of <object>>	Watched applications.	Optional
<i>TMWidget.criteria.applications [CAApplication]</i>	<object>	One CAApplication object.	Optional
<i>TMWidget.criteria.applications [CAApplication].code</i>	<string>	Application code.	Optional
<i>TMWidget.criteria.applications [CAApplication].name</i>	<string>	Application name.	Optional
<i>TMWidget.criteria.applications [CAApplication].tunneled</i>	<string>	Flag: is the application tunneled.	Optional
<i>TMWidget.title</i>	<string>	Widget title.	
<i>TMWidget.attributes</i>	<object>	Widget common attributes.	Optional
<i>TMWidget.attributes.pan_zoomable</i>	<string>	Flag making the graph interactive.	Optional
<i>TMWidget.attributes.line_scale</i>	<string>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.attributes.format_bytes</i>	<string>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, KBYTES, MBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.attributes.show_images</i>	<string>	Flag showing images in a connection graph.	Optional
<i>TMWidget.attributes.open_nodes</i>	<array of <string>>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.attributes.open_nodes[item]</i>	<string>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.attributes.line_style</i>	<string>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.attributes.layout</i>	<string>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.attributes.width</i>	<number>	Widget width.	Optional
<i>TMWidget.attributes.height</i>	<number>	Widget height.	Optional
<i>TMWidget.attributes.percent_of_total</i>	<string>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.attributes.edge_thickness</i>	<string>	Widget edge thickness.	Optional
<i>TMWidget.attributes.display_host_group_type</i>	<string>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.attributes.extend_to_zero</i>	<string>	Flag: extending the Y-axis to zero.	Optional

<i>TMWidget.attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.user_attributes</i>	<i>&lt;object&gt;</i>	User-specific attributes.	Optional
<i>TMWidget.user_attributes.pan_zoomable</i>	<i>&lt;string&gt;</i>	Flag making the graph interactive.	Optional
<i>TMWidget.user_attributes.line_scale</i>	<i>&lt;string&gt;</i>	Line scale for a line chart (can be: LINEAR, LOG).	Optional; Values: LINEAR, LOG
<i>TMWidget.user_attributes.format_bytes</i>	<i>&lt;string&gt;</i>	What unit to use for formatting traffic values (BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF).	Optional; Values: BITS, BYTES, KBYTES, MBYTES, GBITS, GBYTES, AUTOBITS, AUTOBYTES, UI_PREF
<i>TMWidget.user_attributes.show_images</i>	<i>&lt;string&gt;</i>	Flag showing images in a connection graph.	Optional
<i>TMWidget.user_attributes.open_nodes</i>	<i>&lt;array of &lt;string&gt;&gt;</i>	List of open node IDs for a tree widget.	Optional
<i>TMWidget.user_attributes.open_nodes [item]</i>	<i>&lt;string&gt;</i>	ID of an expanded nodes in a tree widget.	Optional
<i>TMWidget.user_attributes.line_style</i>	<i>&lt;string&gt;</i>	Line chart style (can be: LINE, STACKED).	Optional; Values: LINE, STACKED
<i>TMWidget.user_attributes.layout</i>	<i>&lt;string&gt;</i>	Connection graph layout type (can be: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC).	Optional; Values: HORIZONTAL_CLISRV, VERTICAL_CLISRV, HORIZONTAL_TREE, VERTICAL_TREE, RADIAL_TREE, SYMMETRIC
<i>TMWidget.user_attributes.width</i>	<i>&lt;number&gt;</i>	Widget width.	Optional
<i>TMWidget.user_attributes.height</i>	<i>&lt;number&gt;</i>	Widget height.	Optional
<i>TMWidget.user_attributes.percent_of_total</i>	<i>&lt;string&gt;</i>	Flag including the 'total' item in a pie chart.	Optional
<i>TMWidget.user_attributes.edge_thickness</i>	<i>&lt;string&gt;</i>	Widget edge thickness.	Optional
<i>TMWidget.user_attributes.display_host_group_type</i>	<i>&lt;string&gt;</i>	Default host grouping type for displaying grouped hosts.	Optional
<i>TMWidget.user_attributes.extend_to_zero</i>	<i>&lt;string&gt;</i>	Flag: extending the Y-axis to zero.	Optional
<i>TMWidget.user_attributes.collapsible</i>	<i>&lt;string&gt;</i>	Flag indicating if the widget is collapsible.	Optional
<i>TMWidget.user_attributes.high_threshold</i>	<i>&lt;string&gt;</i>	High threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.n_items</i>	<i>&lt;number&gt;</i>	Maximum number of items shown.	Optional
<i>TMWidget.user_attributes.colspan</i>	<i>&lt;number&gt;</i>	How many columns the widget occupies in layout.	Optional
<i>TMWidget.user_attributes.low_threshold</i>	<i>&lt;string&gt;</i>	Low threshold on the chart (in bytes).	Optional
<i>TMWidget.user_attributes.moveable_nodes</i>	<i>&lt;string&gt;</i>	Flag allowing the user to reposition nodes in a connection graph.	Optional
<i>TMWidget.user_attributes.orientation</i>	<i>&lt;string&gt;</i>	Bar chart orientation (can be: VERTICAL, HORIZONTAL).	Optional; Values: VERTICAL, HORIZONTAL
<i>TMWidget.user_attributes.modal_links</i>	<i>&lt;number&gt;</i>	Flag adding modal links on a widget.	Optional
<i>TMWidget.timestamp</i>	<i>&lt;string&gt;</i>	Widget time stamp specification.	Optional

## Reporting: Get report queries

Get information for all queries run as part of this report. Each query has a list of columns.

```
GET https://{device}/api/profiler/1.2/reporting/reports/{report_id}/queries
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "metric": string,
    "actual_log": string
```

```

    actual_log : string,
    "columns": [
    {
      "metric": string,
      "cli_srv": string,
      "comparison_parameter": string,
      "internal": string,
      "id": number,
      "strid": string,
      "statistic": string,
      "severity": string,
      "role": string,
      "category": string,
      "name": string,
      "comparison": string,
      "sortable": string,
      "type": string,
      "direction": string,
      "available": string,
      "context": string,
      "area": string,
      "has_others": string,
      "unit": string,
      "name_type": string,
      "rate": string
    }
  ],
  "id": string,
  "statistic": string,
  "role": string,
  "group_by": string,
  "actual_t0": number,
  "parent_id": string,
  "actual_t1": number,
  "type": string,
  "sort_col": number,
  "direction": string,
  "sort_desc": string,
  "area": string,
  "unit": string,
  "rate": string
}
]

```

Example:

```

[
  {
    "direction": "none",
    "actual_log": "flow",
    "actual_t0": 1352319840,
    "sort_desc": true,
    "area": "none",
    "metric": "none",
    "sort_col": 33,
    "parent_id": "",
    "rate": "none",
    "group_by": "hos",
    "role": "none",
    "columns": [
      {
        "strid": "ID_AVG_BYTES",
        "metric": "net_bw",
        "rate": "persec",
        "statistic": "avg",
        "id": 33,
        "unit": "bytes",
        "category": "data",
        "severity": "none",
        "area": "none",
        "internal": false,
        "role": "none",
        "cli_srv": "none",
        "type": "float",
        "available": true,
        "direction": "none",
        "comparison": "none",
        "sortable": true,
        "name": "Avg Bytes/s",
        "comparison_parameter": "",
        "has_others": false,
        "context": false,
        "name_type": "colname_parts"
      },
      {
        "strid": "ID_AVG_BYTES_RTX",
        "metric": "rtx",
        "rate": "persec",
        "statistic": "avg",

```

```

    "id": 391,
    "unit": "bytes",
    "category": "data",
    "severity": "none",
    "area": "none",
    "internal": false,
    "role": "none",
    "cli_srv": "none",
    "type": "float",
    "available": false,
    "direction": "none",
    "comparison": "none",
    "sortable": true,
    "name": "Avg Retrans Bytes/s",
    "comparison_parameter": "",
    "has_others": false,
    "context": false,
    "name_type": "colname_parts"
  }
],
"statistic": "none",
"type": "summary",
"id": "0:sum_hos_non_non_non_non_non_non_33_d_0",
"unit": "none",
"actual_t1": 1352320200
}
]

```

Property Name	Type	Description	Notes
<i>Queries</i>	<array of <object>>	List of queries. Query is one tabular unit of report data.	
<i>Queries[Query]</i>	<object>	A query.	Optional
<i>Queries[Query].metric</i>	<string>	Query 'metric'. See 'reporting/metrics'.	
<i>Queries[Query].actual_log</i>	<string>	Type of data log file that was used to get data for the query.	
<i>Queries[Query].columns</i>	<array of <object>>	List of columns that constitute the query. See 'reporting/columns'.	
<i>Queries[Query].columns[Column]</i>	<object>	A column for reporting query.	Optional
<i>Queries[Query].columns[Column].metric</i>	<string>	Column 'metric'. See 'reporting/metrics'.	
<i>Queries[Query].columns[Column].cli_srv</i>	<string>	Text flag indicating if the column is for the clients or servers.	
<i>Queries[Query].columns[Column].comparison_parameter</i>	<string>	Parameter for column comparison.	
<i>Queries[Query].columns[Column].internal</i>	<string>	Boolean flag indicating if the column is internal to the system.	
<i>Queries[Query].columns[Column].id</i>	<number>	System ID for the column. Used in the API.	
<i>Queries[Query].columns[Column].strid</i>	<string>	String ID for the column. Not used by the API, but easier for the human user to see.	
<i>Queries[Query].columns[Column].statistic</i>	<string>	Column 'statistic'. See 'reporting/statistics'.	
<i>Queries[Query].columns[Column].severity</i>	<string>	Column 'severity'. See 'reporting/severities'.	
<i>Queries[Query].columns[Column].role</i>	<string>	Column 'role'. See 'reporting/roles'.	
<i>Queries[Query].columns[Column].category</i>	<string>	Column 'category'. See 'reporting/categories'.	
<i>Queries[Query].columns[Column].name</i>	<string>	Column name. Format used for column names is similar to the format used for column data.	
<i>Queries[Query].columns[Column].comparison</i>	<string>	Column 'comparison'. See 'reporting/comparisons'.	
<i>Queries[Query].columns[Column].sortable</i>	<string>	Boolean flag indicating if this data can be sorted on this column when running the template.	
<i>Queries[Query].columns[Column].type</i>	<string>	Type of the column data. See 'reporting/types'.	
<i>Queries[Query].columns[Column].direction</i>	<string>	Column 'direction'. See 'reporting/directions'.	
<i>Queries[Query].columns[Column].available</i>	<string>	Boolean flag indicating that the data for the column is available without the need to re-run the template.	
<i>Queries[Query].columns[Column].context</i>	<string>	Internal flag used for formatting certain kinds of data.	
<i>Queries[Query].columns[Column].area</i>	<string>	Column 'area'. See 'reporting/area'.	
<i>Queries[Query].columns[Column].has_others</i>	<string>	Boolean flag indicating if the column's 'other' row can be computed.	
<i>Queries[Query].columns[Column].unit</i>	<string>	Column 'unit'. See 'reporting/units'.	
<i>Queries[Query].columns[Column].name_type</i>	<string>	Type of the column name. See 'reporting/types'.	
<i>Queries[Query].columns[Column].rate</i>	<string>	Column 'rate'. See 'reporting/rates'.	

<code>Queries[Query].id</code>	<code>&lt;string&gt;</code>	ID for the query. Used in the API.	
<code>Queries[Query].statistic</code>	<code>&lt;string&gt;</code>	Query 'statistic'. See 'reporting/statistics'.	
<code>Queries[Query].role</code>	<code>&lt;string&gt;</code>	Query 'role'. See 'reporting/roles'.	
<code>Queries[Query].group_by</code>	<code>&lt;string&gt;</code>	Grouping of data in the query. See 'reporting/group_bys'.	
<code>Queries[Query].actual_t0</code>	<code>&lt;number&gt;</code>	Actual start time for data in the query. This could be different from the requested start time because of time interval snapping and other similar features.	
<code>Queries[Query].parent_id</code>	<code>&lt;string&gt;</code>	Query ID of the query that preceeded this query and influenced data selection for it. For example, if one runs a query that returns time-series data for top 10 protocols in the network, the first query that would need to run is the one to pick top 10 protocols. That query would be the parent one to the follow-up query to get time-series data for those selected 10 protocols.	
<code>Queries[Query].actual_t1</code>	<code>&lt;number&gt;</code>	Actual end time for the data in the query. See 'actual_t0' for more detail.	
<code>Queries[Query].type</code>	<code>&lt;string&gt;</code>	Internal value. Reserved.	
<code>Queries[Query].sort_col</code>	<code>&lt;number&gt;</code>	ID of that column that was used to sort the query when it ran.	
<code>Queries[Query].direction</code>	<code>&lt;string&gt;</code>	Query 'direction'. See 'reporting/directions'.	
<code>Queries[Query].sort_desc</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if the sorting was in the descending order.	
<code>Queries[Query].area</code>	<code>&lt;string&gt;</code>	Query 'area'. See 'reporting/areas'.	
<code>Queries[Query].unit</code>	<code>&lt;string&gt;</code>	Query 'unit'. See 'reporting/units'.	
<code>Queries[Query].rate</code>	<code>&lt;string&gt;</code>	Query 'rate'. See 'reporting/rates'.	

## Reporting: Create report

Generate a new report.

POST <https://{device}/api/profiler/1.2/reporting/reports>

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```

{
  "criteria": {
    "traffic_expression": string,
    "time_frame": {
      "resolution": string,
      "end": number,
      "expression": string,
      "start": number
    },
    "query": {
      "columns": [
        number
      ],
      "role": string,
      "group_by": string,
      "host_group_type": string,
      "direction": string,
      "sort_column": number,
      "area": string,
      "centricity": string,
      "realm": string
    },
    "deprecated": {
      [prop]: string
    }
  },
  "name": string,
  "template_id": number
}

```

Example:

```

{
  "criteria": {
    "traffic_expression": "app WEB",
    "time_frame": {
      "start": 1352314764,
      "end": 1352315064
    },
    "query": {
      "realm": "traffic_summary",
      "sort_column": 33,
      "group_by": "hos",
      "columns": [
        6,
        33,
        34
      ]
    }
  },
  "template_id": 184,
  "name": "Bytes and packets for the top 20 hosts using application WEB"
}

```

Property Name	Type	Description	Notes
ReportInputs	<object>	ReportInputs object.	
ReportInputs.criteria	<object>	Criteria needed to run the report.	Optional
ReportInputs.criteria.traffic_expression	<string>	Traffic expression.	Optional
ReportInputs.criteria.time_frame	<object>	Time frame object.	Optional
ReportInputs.criteria.time_frame.resolution	<string>	Report data resolution. It can be one of: 1min, 15min, hour, 6hour, day, week, month. If not specified a resolution will automatically be selected based on time frame of the report.	Optional
ReportInputs.criteria.time_frame.end	<number>	Report end time (unix time).	Optional
ReportInputs.criteria.time_frame.expression	<string>	Traffic expression.	Optional
ReportInputs.criteria.time_frame.start	<number>	Report start time (unix time).	Optional
ReportInputs.criteria.query	<object>	Query object.	Optional
ReportInputs.criteria.query.columns	<array of <number>>	Query columns. Can be many of GET /reporting/columns.	Optional
ReportInputs.criteria.query.columns [item]	<number>	Query column.	Optional
ReportInputs.criteria.query.role	<string>	Query role. Can be one of /reporting/roles.	Optional
ReportInputs.criteria.query.group_by	<string>	Query group_by. Can be one of GET /reporting/group_bys.	Optional
ReportInputs.criteria.query.host_group_type	<string>	Query host group type. Required for "host group (gro)" "host group pairs (gpp)" and "host group pairs with ports (gpr)" queries.	Optional



<i>ReportInputs.criteria.query.direction</i>	<string>	Query direction. Can be one of GET /reporting/directions.	Optional
<i>ReportInputs.criteria.query.sort_column</i>	<number>	Query sort column. Can be one of GET /reporting/columns.	Optional
<i>ReportInputs.criteria.query.area</i>	<string>	Query area. Can be one of GET /reporting/areas.	Optional
<i>ReportInputs.criteria.query.centricity</i>	<string>	Query centricity. Can be one of GET /reporting/centricities.	Optional
<i>ReportInputs.criteria.query.realm</i>	<string>	Query realm. Can be one of GET /reporting/realm.	
<i>ReportInputs.criteria.deprecated</i>	<object>	Map with legacy criteria attributes that will not be supported soon.	Optional
<i>ReportInputs.criteria.deprecated[prop]</i>	<string>	ReportDeprecatedFilters map value.	Optional
<i>ReportInputs.name</i>	<string>	Report name.	Optional
<i>ReportInputs.template_id</i>	<number>	Template ID. Can be one of GET /reporting/templates.	

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "run_time": number,
  "error_text": string,
  "remaining_seconds": number,
  "saved": string,
  "id": number,
  "status": string,
  "percent": number,
  "user_id": number,
  "size": number,
  "name": string,
  "template_id": number
}
```

Example:

```
{
  "status": "completed",
  "user_id": 1,
  "name": "Host Information Report",
  "percent": 100,
  "id": 1001,
  "remaining_seconds": 0,
  "run_time": 1352494550,
  "saved": true,
  "template_id": 952,
  "error_text": "",
  "size": 140
}
```

Property Name	Type	Description	Notes
<i>ReportInfo</i>	<object>	Object representing report information.	
<i>ReportInfo.run_time</i>	<number>	Time when the report was run (Unix time).	
<i>ReportInfo.error_text</i>	<string>	A report can be completed with an error. Error message may provide more detailed info.	Optional
<i>ReportInfo.remaining_seconds</i>	<number>	Number of seconds remaining to run the report. Even if this number is 0, the report may not yet be completed, so check 'status' to make sure what the status is.	
<i>ReportInfo.saved</i>	<string>	Boolean flag indicating if the report was saved.	
<i>ReportInfo.id</i>	<number>	ID of the report. To be used in the API.	
<i>ReportInfo.status</i>	<string>	Status of the report.	Values: completed, running, waiting
<i>ReportInfo.percent</i>	<number>	Progress of the report represented by percentage of report completion.	
<i>ReportInfo.user_id</i>	<number>	ID of the user who owns the report.	
<i>ReportInfo.size</i>	<number>	Size of the report in kilobytes.	
<i>ReportInfo.name</i>	<string>	Name of the report. Could be given by a user or automatically generated by the system.	Optional
<i>ReportInfo.template_id</i>	<number>	ID of the template that the report is based on.	

## Host\_Group\_Types: Get host group members

Get a list of hosts in a specified host group.

```
GET https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}/groups/{group_id}/members?offset={number}&sort={string}&limit={number}
```

## Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "idaddr": string
  }
]
```

Example:

```
[
  {
    "idaddr": "10.99.16.41"
  },
  {
    "idaddr": "10.99.16.43"
  }
]
```

Property Name	Type	Description	Notes
<i>HostGroupMembers</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of host group members.	
<i>HostGroupMembers[HostGroupMember]</i>	<i>&lt;object&gt;</i>	Object representing a host group member.	Optional
<i>HostGroupMembers[HostGroupMember].idaddr</i>	<i>&lt;string&gt;</i>	Host group member's IP address.	

## Host\_Group\_Types: Get host group type

Get one host grouping type.

```
GET https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "favorite": string,
  "id": number,
  "description": string,
  "name": string,
  "type": string
}
```

Example:

```
{
  "favorite": true,
  "description": "Groups based on the location of their member hosts, such as NY, Dallas, DataCenter1, etc.",
  "type": "User-created",
  "name": "ByLocation",
  "id": 102
}
```

Property Name	Type	Description	Notes
<i>HostGroupType</i>	<object>	Object representing a host group type.	
<i>HostGroupType.favorite</i>	<string>	If the host group type is favorite.	
<i>HostGroupType.id</i>	<number>	Host group type's ID.	
<i>HostGroupType.description</i>	<string>	Host group type's description.	
<i>HostGroupType.name</i>	<string>	Host group type's name.	
<i>HostGroupType.type</i>	<string>	Host group type's type.	Values: User-created, System-created

## Host\_Group\_Types: Get host group

Get one host group.

```
GET https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}/groups/{group_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "id": number,
  "name": string
}
```

Example:

```
{
  "name": "Columbus",
  "id": 6
}
```

Property Name	Type	Description	Notes
<i>HostGroup</i>	<object>	Object representing a host group.	
<i>HostGroup.id</i>	<number>	Host group's ID.	
<i>HostGroup.name</i>	<string>	Host group's name.	

## Host\_Group\_Types: Delete host group type

Delete one host grouping type.

```
DELETE https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Host\_Group\_Types: List host group types

Get a list of host grouping types.

```
GET https://{device}/api/profiler/1.2/host_group_types?favorite={string}&offset={number}&sortBy={string}&sort={string}&limit={number}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>favorite</i>	<i>&lt;string&gt;</i>	Show only host grouping types with specific state of the 'favorite' flag.	Optional
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>sortBy</i>	<i>&lt;string&gt;</i>	Sorting field name.	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>type</i>	<i>&lt;string&gt;</i>	Show only host grouping types of type: 'User-created' or 'System-created'.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
[
  {
    "favorite": string,
    "id": number,
    "description": string,
    "name": string,
    "type": string
  }
]
```

Example:

```
[
  {
    "favorite": true,
    "description": "Groups based on the function of their member hosts, such as Email, Web, etc. ",
    "type": "User-created",
    "name": "ByFunction",
    "id": 100
  },
  {
    "favorite": true,
    "description": "Groups based on the location of their member hosts, such as NY, Dallas, DataCenter1, etc.",
    "type": "User-created",
    "name": "ByLocation",
    "id": 102
  }
]
```

Property Name	Type	Description	Notes
<i>HostGroupTypes</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of host group types.	
<i>HostGroupTypes[HostGroupType]</i>	<i>&lt;object&gt;</i>	Object representing a host group type.	Optional
<i>HostGroupTypes[HostGroupType].favorite</i>	<i>&lt;string&gt;</i>	If the host group type is favorite.	
<i>HostGroupTypes[HostGroupType].id</i>	<i>&lt;number&gt;</i>	Host group type's ID.	
<i>HostGroupTypes[HostGroupType].description</i>	<i>&lt;string&gt;</i>	Host group type's description.	
<i>HostGroupTypes[HostGroupType].name</i>	<i>&lt;string&gt;</i>	Host group type's name.	
<i>HostGroupTypes[HostGroupType].type</i>	<i>&lt;string&gt;</i>	Host group type's type.	Values: User-created, System-created

## Host\_Group\_Types: List host groups

Get a list of host groups for a given host grouping type.

```
GET https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}/groups?offset={number}&sortBy={string}&sort={string}&limit={number}
```

## Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>sortBy</i>	<i>&lt;string&gt;</i>	Sorting field name (default: 'name').	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "id": number,
    "name": string
  }
]
```

Example:

```
[
  {
    "name": "Austin",
    "id": 13
  },
  {
    "name": "Columbus",
    "id": 6
  }
]
```

Property Name	Type	Description	Notes
<i>HostGroups</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of host groups.	
<i>HostGroup[HostGroup]</i>	<i>&lt;object&gt;</i>	Object representing a host group.	Optional
<i>HostGroup[HostGroup].id</i>	<i>&lt;number&gt;</i>	Host group's ID.	
<i>HostGroup[HostGroup].name</i>	<i>&lt;string&gt;</i>	Host group's name.	

## Host\_Group\_Types: Get host group type config

Get host grouping type configuration.

```
GET https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}/config
```

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "name": string,
    "cidr": string
  }
]
```

Example:

```
[
  {
    "cidr": "10.99.11.0/255.255.255.0",
    "name": "Seattle"
  },
  {
    "cidr": "10.99.12.0/255.255.255.0",
    "name": "LosAngeles"
  }
]
```

Property Name	Type	Description	Notes
<i>HostGroupTypeDefs</i>	<array of <object>>	List of host group type definitions.	
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}	<object>	Object representing host group type's definition.	Optional
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}.name	<string>	Host group type definition's name.	
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}.cidr	<string>	Host group type definition's CIDRs.	

## Host\_Group\_Types: Create host group type

Create a new host grouping type.

POST https://{device}/api/profiler/1.2/host\_group\_types

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```
{
  "favorite": string,
  "config": [
    {
      "name": string,
      "cidr": string
    }
  ],
  "description": string,
  "name": string
}
```

Example:

```
{
  "favorite": true,
  "name": "ByLocation",
  "description": "Groups based on the location of their member hosts, such as NY, Dallas, DataCenter1, etc."
}
```

Property Name	Type	Description	Notes
<i>HostGroupTypeUp</i>	<object>	Object representing a new host group type.	
<i>HostGroupTypeUp</i> .favorite	<string>	If new host group type is favorite.	
<i>HostGroupTypeUp</i> .config	<array of <object>>	Optional configuration of the grouptype.	Optional
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef]	<object>	Object representing host group type's definition.	Optional
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef].name	<string>	Host group type definition's name.	
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef].cidr	<string>	Host group type definition's CIDRs.	
<i>HostGroupTypeUp</i> .description	<string>	New host group type's description.	

<code>HostGroupTypeUp.name</code>	<code>&lt;string&gt;</code>	New host group type's name.	
-----------------------------------	-----------------------------	-----------------------------	--

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
{
  "favorite": string,
  "config": [
    {
      "name": string,
      "cidr": string
    }
  ],
  "description": string,
  "name": string
}
```

Example:

```
{
  "favorite": true,
  "name": "ByLocation",
  "description": "Groups based on the location of their member hosts, such as NY, Dallas, DataCenter1, etc."
}
```

Property Name	Type	Description	Notes
<code>HostGroupTypeUp</code>	<code>&lt;object&gt;</code>	Object representing a new host group type.	
<code>HostGroupTypeUp.favorite</code>	<code>&lt;string&gt;</code>	If new host group type is favorite.	
<code>HostGroupTypeUp.config</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Optional configuration of the grouptype.	Optional
<code>HostGroupTypeUp.config[HostGroupTypeDef]</code>	<code>&lt;object&gt;</code>	Object representing host group type's definition.	Optional
<code>HostGroupTypeUp.config[HostGroupTypeDef].name</code>	<code>&lt;string&gt;</code>	Host group type definition's name.	
<code>HostGroupTypeUp.config[HostGroupTypeDef].cidr</code>	<code>&lt;string&gt;</code>	Host group type definition's CIDRs.	
<code>HostGroupTypeUp.description</code>	<code>&lt;string&gt;</code>	New host group type's description.	
<code>HostGroupTypeUp.name</code>	<code>&lt;string&gt;</code>	New host group type's name.	

## Host\_Group\_Types: Set host group type config

Update host grouping type configuration.

```
PUT https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}/config
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
[
  {
    "name": string,
    "cidr": string
  }
]
```

Example:

```
[
  {
    "cidr": "10.99.11.0/255.255.255.0",
    "name": "Seattle"
  },
  {
    "cidr": "10.99.12.0/255.255.255.0",
    "name": "LosAngeles"
  }
]
```

Property Name	Type	Description	Notes
<i>HostGroupTypeDefs</i>	<array of <object>>	List of host group type definitions.	
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}	<object>	Object representing host group type's definition.	Optional
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}.name	<string>	Host group type definition's name.	
<i>HostGroupTypeDefs</i> {HostGroupTypeDef}.cidr	<string>	Host group type definition's CIDRs.	

## Response Body

On success, the server does not provide any body in the responses.

## Host\_Group\_Types: Update host group type

Update one host grouping type.

```
PUT https://{device}/api/profiler/1.2/host_group_types/{host_group_type_id}
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
{
  "favorite": string,
  "config": [
    {
      "name": string,
      "cidr": string
    }
  ],
  "description": string,
  "name": string
}
```

Example:

```
{
  "favorite": true,
  "name": "ByLocation",
  "description": "Groups based on the location of their member hosts, such as NY, Dallas, DataCenter1, etc."
}
```

Property Name	Type	Description	Notes
<i>HostGroupTypeUp</i>	<object>	Object representing a new host group type.	
<i>HostGroupTypeUp</i> .favorite	<string>	If new host group type is favorite.	
<i>HostGroupTypeUp</i> .config	<array of <object>>	Optional configuration of the grouptype.	Optional
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef]	<object>	Object representing host group type's definition.	Optional
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef].name	<string>	Host group type definition's name.	
<i>HostGroupTypeUp</i> .config[HostGroupTypeDef].cidr	<string>	Host group type definition's CIDRs.	
<i>HostGroupTypeUp</i> .description	<string>	New host group type's description.	
<i>HostGroupTypeUp</i> .name	<string>	New host group type's name.	

## Response Body

On success, the server does not provide any body in the responses.

## Applications: Get application

Get configuration of a specific application.

```
GET https://{device}/api/profiler/1.2/applications/{application_id}
```

## Authorization

This request requires authorization.



## Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "enabled": string,
  "id": number,
  "name": string,
  "type": string,
  "priority": number,
  "sources": [
    {
      "id": number
    }
  ]
}
```

Example:

```
{
  "name": "LDAP",
  "enabled": true,
  "priority": 4,
  "sources": [
    {
      "id": 1
    },
    {
      "id": 0
    },
    {
      "id": 2
    },
    {
      "id": 100
    }
  ],
  "type": "Layer_7",
  "id": 4
}
```

Property Name	Type	Description	Notes
<i>Application</i>	<i>&lt;object&gt;</i>	Object representing an application.	
<i>Application.enabled</i>	<i>&lt;string&gt;</i>	if the application enabled.	
<i>Application.id</i>	<i>&lt;number&gt;</i>	Application's ID.	
<i>Application.name</i>	<i>&lt;string&gt;</i>	Application's name.	
<i>Application.type</i>	<i>&lt;string&gt;</i>	Application's type.	Values: Layer_7, Layer_4
<i>Application.priority</i>	<i>&lt;number&gt;</i>	Application's priority.	
<i>Application.sources</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Application's sources.	
<i>Application.sources[ApplicationSourceId]</i>	<i>&lt;object&gt;</i>	Object representing an application source ID.	Optional
<i>Application.sources[ApplicationSourceId].id</i>	<i>&lt;number&gt;</i>	ID of application source's ID.	

## Applications: Get application sources

List all application sources.

```
GET https://{device}/api/profiler/1.2/applications/sources?offset={number}&limit={number}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

## JSON

```
[
  {
    "id": number,
    "name": string
  }
]
```

Example:

```
[
  {
    "name": "Sensor",
    "id": 100
  },
  {
    "name": "Shark",
    "id": 3
  },
  {
    "name": "Steelhead",
    "id": 2
  }
]
```

Property Name	Type	Description	Notes
<i>ApplicationSources</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of application sources.	
<i>ApplicationSources[ApplicationSource]</i>	<i>&lt;object&gt;</i>	Object representing an application sources.	Optional
<i>ApplicationSources[ApplicationSource].id</i>	<i>&lt;number&gt;</i>	Application source's ID.	
<i>ApplicationSources[ApplicationSource].name</i>	<i>&lt;string&gt;</i>	Application source's name.	

## Applications: List layer 4 applications

Get a list of all supported Layer 4 applications.

```
GET https://{device}/api/profiler/1.2/applications/layer_4?enabled={string}&offset={number}&orderby={string}&sort={string}&override={string}&priority={number}&limit={number}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>enabled</i>	<i>&lt;string&gt;</i>	Show only enabled or disabled application (use 'true' or 'false').	Optional
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>orderby</i>	<i>&lt;string&gt;</i>	Sorting field name.	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>override</i>	<i>&lt;string&gt;</i>	Filter applications by the 'override' class.	Optional
<i>priority</i>	<i>&lt;number&gt;</i>	Filter applications by a priority.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

### Response Body

On success, the server returns a response body with the following structure:

## JSON

```
[
  {
    "enabled": string,
    "config": {
      "port_groups": [
        {
          "name": string,
          "group_id": number
        }
      ],
      "proto_ports": [
        {
          "port": number
        }
      ]
    }
  }
]
```

```

    "port": number,
    "protocol": number,
    "name": string
  }
],
"hosts": [
  string
],
"cidrs": [
  string
],
"host_groups": [
  {
    "host_group": number,
    "name": string,
    "host_group_type": number
  }
],
"override": string,
"app_id": number,
"priority": number
}
]

```

Example:

```

[
  {
    "name": "CIFS",
    "enabled": true,
    "app_id": 144,
    "priority": 1,
    "override": "Unknown",
    "config": {
      "cidrs": [
        "0.0.0.0/0"
      ],
      "proto_ports": [
        {
          "protocol": 6,
          "name": "tcp/139(netbios-ssn)",
          "port": 139
        },
        {
          "protocol": 6,
          "name": "tcp/445(microsoft-ds)",
          "port": 445
        }
      ]
    }
  },
  {
    "name": "FTP",
    "enabled": true,
    "app_id": 2,
    "priority": 2,
    "override": "Always",
    "config": {
      "cidrs": [
        "0.0.0.0/0"
      ],
      "proto_ports": [
        {
          "protocol": 6,
          "name": "tcp/20(ftp-data)",
          "port": 20
        },
        {
          "protocol": 6,
          "name": "tcp/21(ftp)",
          "port": 21
        }
      ]
    }
  }
]
]

```

Property Name	Type	Description	Notes
<i>ApplicationsLayer4</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of all mapped applications.	
<i>ApplicationsLayer4[ApplicationLayer4]</i>	<i>&lt;object&gt;</i>	Object representing a mapped application.	Optional

<code>ApplicationsLayer4[ApplicationLayer4].enabled</code>	<code>&lt;string&gt;</code>	If the mapped application is enabled.	
<code>ApplicationsLayer4[ApplicationLayer4].config</code>	<code>&lt;object&gt;</code>	Object representing mapped application's configuration.	
<code>ApplicationsLayer4[ApplicationLayer4].config.port_groups</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Object representing mapped application's port_groups.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.port_groups [ApplicationPortGroup]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's port group.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.port_groups [ApplicationPortGroup].name</code>	<code>&lt;string&gt;</code>	Mapped application's port group name.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.port_groups [ApplicationPortGroup].group_id</code>	<code>&lt;number&gt;</code>	Mapped application's port group ID.	
<code>ApplicationsLayer4[ApplicationLayer4].config.proto_ports</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Object representing mapped application's proto_ports.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.proto_ports [ApplicationProtoPort]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's proto port.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.proto_ports [ApplicationProtoPort].port</code>	<code>&lt;number&gt;</code>	Mapped application's port.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.proto_ports [ApplicationProtoPort].protocol</code>	<code>&lt;number&gt;</code>	Mapped application's protocol.	
<code>ApplicationsLayer4[ApplicationLayer4].config.proto_ports [ApplicationProtoPort].name</code>	<code>&lt;string&gt;</code>	Mapped application's proto port name.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.hosts</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's hosts.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.hosts[item]</code>	<code>&lt;string&gt;</code>	Mapped application's host.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.cidrs</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's CIDRs.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.cidrs[item]</code>	<code>&lt;string&gt;</code>	Mapped application's CIDR.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.host_groups</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Object representing mapped application's host_groups.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.host_groups [ApplicationHostGroup]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's host group.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.host_groups [ApplicationHostGroup].host_group</code>	<code>&lt;number&gt;</code>	Mapped application's host group.	
<code>ApplicationsLayer4[ApplicationLayer4].config.host_groups [ApplicationHostGroup].name</code>	<code>&lt;string&gt;</code>	Mapped application's host group's name.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].config.host_groups [ApplicationHostGroup].host_group_type</code>	<code>&lt;number&gt;</code>	Mapped application's host group type.	
<code>ApplicationsLayer4[ApplicationLayer4].id</code>	<code>&lt;number&gt;</code>	Mapped application's ID.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].name</code>	<code>&lt;string&gt;</code>	Mapped application's name.	
<code>ApplicationsLayer4[ApplicationLayer4].override</code>	<code>&lt;string&gt;</code>	Mapped application's override.	Values: Always, Unclassified, Unknown
<code>ApplicationsLayer4[ApplicationLayer4].app_id</code>	<code>&lt;number&gt;</code>	Application ID.	Optional
<code>ApplicationsLayer4[ApplicationLayer4].priority</code>	<code>&lt;number&gt;</code>	Mapped application's priority.	Optional

## Applications: List applications

Get a list of all supported Layer 4 and Layer 7 applications.

```
GET https://{device}/api/profiler/1.2/applications?enabled={string}&offset={number}&sortby={string}&sort={string}&type={string}&sources={string}&limit={number}
```

### Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>enabled</i>	<i>&lt;string&gt;</i>	Show only enabled or disabled applications (use 'true' or 'false').	Optional
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>sortby</i>	<i>&lt;string&gt;</i>	Sorting field name (default: 'name').	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>type</i>	<i>&lt;string&gt;</i>	Filter applications by application type ('Layer_4' or 'Layer_7').	Optional
<i>sources</i>	<i>&lt;string&gt;</i>	Filter the applications by source ID.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
[
  {
    "enabled": string,
    "id": number,
    "name": string,
    "type": string,
    "priority": number,
    "sources": [
      {
        "id": number
      }
    ]
  }
]
```

Example:

```
[
  {
    "name": "PCMA",
    "enabled": true,
    "priority": 399,
    "sources": [
      {
        "id": 100
      }
    ],
    "type": "Layer_7",
    "id": 399
  },
  {
    "name": "LDAP",
    "enabled": true,
    "priority": 4,
    "sources": [
      {
        "id": 1
      },
      {
        "id": 0
      },
      {
        "id": 2
      },
      {
        "id": 100
      }
    ],
    "type": "Layer_7",
    "id": 4
  }
]
```

Property Name	Type	Description	Notes
<i>Applications</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of applications.	
<i>Applications[Application]</i>	<i>&lt;object&gt;</i>	Object representing an application.	Optional
<i>Applications[Application].enabled</i>	<i>&lt;string&gt;</i>	if the application enabled.	
<i>Applications[Application].id</i>	<i>&lt;number&gt;</i>	Application's ID.	
<i>Applications[Application].name</i>	<i>&lt;string&gt;</i>	Application's name.	

---

<i>Applications</i> [Application].type	<string>	Application's type.	Values: Layer_7, Layer_4
<i>Applications</i> [Application].priority	<number>	Application's priority.	
<i>Applications</i> [Application].sources	<array of <object>>	Application's sources.	
<i>Applications</i> [Application].sources [ApplicationSourceId]	<object>	Object representing an application source ID.	Optional
<i>Applications</i> [Application].sources [ApplicationSourceId].id	<number>	ID of application source's ID.	

---

## Applications: Get layer 4 application

Get information for a Layer 4 application.

```
GET https://{device}/api/profiler/1.2/applications/layer_4/{layer4_app_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```

{
  "enabled": string,
  "config": {
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "proto_ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "hosts": [
      string
    ],
    "cidrs": [
      string
    ],
    "host_groups": [
      {
        "host_group": number,
        "name": string,
        "host_group_type": number
      }
    ]
  },
  "id": number,
  "name": string,
  "override": string,
  "app_id": number,
  "priority": number
}

```

Example:

```

{
  "name": "CIFS",
  "enabled": true,
  "app_id": 144,
  "priority": 1,
  "override": "Unknown",
  "config": {
    "cidrs": [
      "0.0.0.0/0"
    ],
    "proto_ports": [
      {
        "protocol": 6,
        "name": "tcp/139(netbios-ssn)",
        "port": 139
      },
      {
        "protocol": 6,
        "name": "tcp/445(microsoft-ds)",
        "port": 445
      }
    ]
  },
  "id": 1
}

```

Property Name	Type	Description	Notes
<i>ApplicationLayer4</i>	<object>	Object representing a mapped application.	
<i>ApplicationLayer4.enabled</i>	<string>	If the mapped application is enabled.	
<i>ApplicationLayer4.config</i>	<object>	Object representing mapped application's configuration.	
<i>ApplicationLayer4.config.port_groups</i>	<array of <object>>	Object representing mapped application's port_groups.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ApplicationPortGroup]	<object>	Object representing one mapped application's port group.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ApplicationPortGroup].name	<string>	Mapped application's port group name.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ApplicationPortGroup].group_id	<number>	Mapped application's port group ID.	
<i>ApplicationLayer4.config.proto_ports</i>	<array of <object>>	Object representing mapped application's proto_ports.	Optional

<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's proto port.	Optional
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].port</code>	<code>&lt;number&gt;</code>	Mapped application's port.	Optional
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].protocol</code>	<code>&lt;number&gt;</code>	Mapped application's protocol.	
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].name</code>	<code>&lt;string&gt;</code>	Mapped application's proto port name.	Optional
<code>ApplicationLayer4.config.hosts</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's hosts.	Optional
<code>ApplicationLayer4.config.hosts[item]</code>	<code>&lt;string&gt;</code>	Mapped application's host.	Optional
<code>ApplicationLayer4.config.cidrs</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's CIDRs.	Optional
<code>ApplicationLayer4.config.cidrs[item]</code>	<code>&lt;string&gt;</code>	Mapped application's CIDR.	Optional
<code>ApplicationLayer4.config.host_groups</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Object representing mapped application's host_groups.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's host group.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].host_group</code>	<code>&lt;number&gt;</code>	Mapped application's host group.	
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].name</code>	<code>&lt;string&gt;</code>	Mapped application's host group's name.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].host_group_type</code>	<code>&lt;number&gt;</code>	Mapped application's host group type.	
<code>ApplicationLayer4.id</code>	<code>&lt;number&gt;</code>	Mapped application's ID.	Optional
<code>ApplicationLayer4.name</code>	<code>&lt;string&gt;</code>	Mapped application's name.	
<code>ApplicationLayer4.override</code>	<code>&lt;string&gt;</code>	Mapped application's override.	Values: Always, Unclassified, Unknown
<code>ApplicationLayer4.app_id</code>	<code>&lt;number&gt;</code>	Application ID.	Optional
<code>ApplicationLayer4.priority</code>	<code>&lt;number&gt;</code>	Mapped application's priority.	Optional

## Applications: Get layer 7 application

Get Layer 7 application information.

```
GET https://{device}/api/profiler/1.2/applications/layer_7/{layer7_app_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON



```

{
  "enabled": string,
  "signatures": [
    {
      "type": string,
      "signature": string
    }
  ],
  "id": number,
  "name": string,
  "sources": [
    {
      "id": number
    }
  ]
}

```

Example:

```

{
  "sources": [
    {
      "id": 0
    },
    {
      "id": 3
    },
    {
      "id": 2
    },
    {
      "id": 100
    }
  ],
  "signatures": [
    {
      "type": "Builtin",
      "signature": "\\GET daap:\\(.|\\x0a)\\iTunes^[0-9]\\.[0-9]\\.[0-9]"
    },
    {
      "type": "Builtin",
      "signature": "\\DAAP-Server: iTunes^[0-9]\\.[0-9]\\.[0-9]"
    }
  ],
  "enabled": true,
  "name": "iTunes",
  "id": 33
}

```

Property Name	Type	Description	Notes
<i>ApplicationLayer7</i>	<object>	Object representing an unmapped application.	
<i>ApplicationLayer7.enabled</i>	<string>	If the unmapped application is enabled.	
<i>ApplicationLayer7.signatures</i>	<array of <object>>	Unmapped application's hosts.	
<i>ApplicationLayer7.signatures</i> [ApplicationSignature]	<object>	Object representing an application host.	Optional
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].type	<string>	Unmapped application's type.	Values: Builtin, URL, String, Hex_String
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].signature	<string>	Unmapped application's signature.	
<i>ApplicationLayer7.id</i>	<number>	Unmapped application's ID.	Optional
<i>ApplicationLayer7.name</i>	<string>	Unmapped application's name.	
<i>ApplicationLayer7.sources</i>	<array of <object>>	Unmapped application's override.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId]	<object>	Object representing an application source ID.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId].id	<number>	ID of application source's ID.	

## Applications: Create layer 7 application

Create a new Layer 7 application.

POST [https://{device}/api/profiler/1.2/applications/layer\\_7](https://{device}/api/profiler/1.2/applications/layer_7)

### Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

### JSON

```
{
  "enabled": string,
  "signatures": [
    {
      "type": string,
      "signature": string
    }
  ],
  "id": number,
  "name": string,
  "sources": [
    {
      "id": number
    }
  ]
}
```

Example:

```
{
  "sources": [
    {
      "id": 0
    },
    {
      "id": 3
    },
    {
      "id": 2
    },
    {
      "id": 100
    }
  ],
  "signatures": [
    {
      "type": "Builtin",
      "signature": "^\\\"GET daap:\\\"(.|\\x0a)\\\"iTunes\\\"[0-9]\\\"\\.\\\"[0-9]\\\"\\.\\\"[0-9]\\\"
    },
    {
      "type": "Builtin",
      "signature": "\\\"DAAP-Server: iTunes\\\"[0-9]\\\"\\.\\\"[0-9]\\\"\\.\\\"[0-9]\\\"
    }
  ],
  "enabled": true,
  "name": "iTunes",
  "id": 33
}
```

Property Name	Type	Description	Notes
<i>ApplicationLayer7</i>	<object>	Object representing an unmapped application.	
<i>ApplicationLayer7.enabled</i>	<string>	If the unmapped application is enabled.	
<i>ApplicationLayer7.signatures</i>	<array of <object>>	Unmapped application's hosts.	
<i>ApplicationLayer7.signatures</i> [ApplicationSignature]	<object>	Object representing an application host.	Optional
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].type	<string>	Unmapped application's type.	Values: Builtin, URL, String, Hex_String
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].signature	<string>	Unmapped application's signature.	
<i>ApplicationLayer7.id</i>	<number>	Unmapped application's ID.	Optional
<i>ApplicationLayer7.name</i>	<string>	Unmapped application's name.	
<i>ApplicationLayer7.sources</i>	<array of <object>>	Unmapped application's override.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId]	<object>	Object representing an application source ID.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId].id	<number>	ID of application source's ID.	

## Response Body

On success, the server returns a response body with the following structure:

## JSON

```
{
  "enabled": string,
  "signatures": [
    {
      "type": string,
      "signature": string
    }
  ],
  "id": number,
  "name": string,
  "sources": [
    {
      "id": number
    }
  ]
}
```

Example:

```
{
  "sources": [
    {
      "id": 0
    },
    {
      "id": 3
    },
    {
      "id": 2
    },
    {
      "id": 100
    }
  ],
  "signatures": [
    {
      "type": "Builtin",
      "signature": "^\\GET daap:\\.(\\.\\|\\x0a)\\|iTunes\\|[0-9]\\|[0-9]\\|[0-9]"
    },
    {
      "type": "Builtin",
      "signature": "\\DAAP-Server: iTunes\\|[0-9]\\|[0-9]\\|[0-9]"
    }
  ],
  "enabled": true,
  "name": "iTunes",
  "id": 33
}
```

Property Name	Type	Description	Notes
<i>ApplicationLayer7</i>	<i>&lt;object&gt;</i>	Object representing an unmapped application.	
<i>ApplicationLayer7.enabled</i>	<i>&lt;string&gt;</i>	If the unmapped application is enabled.	
<i>ApplicationLayer7.signatures</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Unmapped application's hosts.	
<i>ApplicationLayer7.signatures [ApplicationSignature]</i>	<i>&lt;object&gt;</i>	Object representing an application host.	Optional
<i>ApplicationLayer7.signatures [ApplicationSignature].type</i>	<i>&lt;string&gt;</i>	Unmapped application's type.	Values: Builtin, URL, String, Hex_String
<i>ApplicationLayer7.signatures [ApplicationSignature].signature</i>	<i>&lt;string&gt;</i>	Unmapped application's signature.	
<i>ApplicationLayer7.id</i>	<i>&lt;number&gt;</i>	Unmapped application's ID.	Optional
<i>ApplicationLayer7.name</i>	<i>&lt;string&gt;</i>	Unmapped application's name.	
<i>ApplicationLayer7.sources</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	Unmapped application's override.	Optional
<i>ApplicationLayer7.sources [ApplicationSourceId]</i>	<i>&lt;object&gt;</i>	Object representing an application source ID.	Optional
<i>ApplicationLayer7.sources [ApplicationSourceId].id</i>	<i>&lt;number&gt;</i>	ID of application source's ID.	

## Applications: List layer 7 applications

Get a list of all supported Layer 7 applications.

```
GET https://{device}/api/profiler/1.2/applications/layer_7?enabled={string}&offset={number}&sortBy={string}&sort={string}&sources={string}&limit={number}
```

## Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>enabled</i>	<i>&lt;string&gt;</i>	Show only enabled or disabled application (use 'true' or 'false').	Optional
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>sortby</i>	<i>&lt;string&gt;</i>	Sorting field name.	Optional
<i>sort</i>	<i>&lt;string&gt;</i>	Sorting direction: 'asc' or 'desc' (default: 'asc').	Optional
<i>sources</i>	<i>&lt;string&gt;</i>	Filter Layer 7 applications to those from specific sources.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```
[
  {
    "enabled": string,
    "signatures": [
      {
        "type": string,
        "signature": string
      }
    ],
    "id": number,
    "name": string,
    "sources": [
      {
        "id": number
      }
    ]
  }
]
```

Example:

```
[
  {
    "sources": [
      {
        "id": 0
      },
      {
        "id": 3
      },
      {
        "id": 2
      },
      {
        "id": 100
      }
    ],
    "signatures": [
      {
        "type": "Builtin",
        "signature": "^\"GET daap:\\"(.|\\x0a)\"iTunes\"[0-9]\\.\"[0-9]\"\\.\"[0-9]\""
      },
      {
        "type": "Builtin",
        "signature": "\"DAAP-Server: iTunes\"[0-9]\\.\"[0-9]\"\\.\"[0-9]\""
      }
    ],
    "enabled": true,
    "name": "iTunes",
    "id": 33
  },
  {
    "sources": [
      {
        "id": 100
      }
    ],
    "signatures": [],
    "enabled": true,
    "name": "PCMA",
    "id": 399
  }
]
```

Property Name	Type	Description	Notes
<i>ApplicationsLayer7</i>	<array of <object>>	List of all unmapped applications.	
<i>ApplicationsLayer7</i> [ApplicationLayer7]	<object>	Object representing an unmapped application.	Optional
<i>ApplicationsLayer7</i> [ApplicationLayer7].enabled	<string>	If the unmapped application is enabled.	
<i>ApplicationsLayer7</i> [ApplicationLayer7].signatures	<array of <object>>	Unmapped application's hosts.	
<i>ApplicationsLayer7</i> [ApplicationLayer7].signatures[ApplicationSignature]	<object>	Object representing an application host.	Optional
<i>ApplicationsLayer7</i> [ApplicationLayer7].signatures[ApplicationSignature].type	<string>	Unmapped application's type.	Values: Builtin, URL, String, Hex_String
<i>ApplicationsLayer7</i> [ApplicationLayer7].signatures[ApplicationSignature].signature	<string>	Unmapped application's signature.	
<i>ApplicationsLayer7</i> [ApplicationLayer7].id	<number>	Unmapped application's ID.	Optional

<code>ApplicationsLayer7[ApplicationLayer7].name</code>	<code>&lt;string&gt;</code>	Unmapped application's name.	
<code>ApplicationsLayer7[ApplicationLayer7].sources</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Unmapped application's override.	Optional
<code>ApplicationsLayer7[ApplicationLayer7].sources[ApplicationSourceId]</code>	<code>&lt;object&gt;</code>	Object representing an application source ID.	Optional
<code>ApplicationsLayer7[ApplicationLayer7].sources[ApplicationSourceId].id</code>	<code>&lt;number&gt;</code>	ID of application source's ID.	

---

## Applications: Create layer 4 application

Create a new Layer 4 application.

```
POST https://{device}/api/profiler/1.2/applications/layer_4
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

```
JSON
```

```

{
  "enabled": string,
  "config": {
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "proto_ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "hosts": [
      string
    ],
    "cidrs": [
      string
    ],
    "host_groups": [
      {
        "host_group": number,
        "name": string,
        "host_group_type": number
      }
    ]
  },
  "id": number,
  "name": string,
  "override": string,
  "app_id": number,
  "priority": number
}

```

Example:

```

{
  "name": "CIFS",
  "enabled": true,
  "app_id": 144,
  "priority": 1,
  "override": "Unknown",
  "config": {
    "cidrs": [
      "0.0.0.0/0"
    ],
    "proto_ports": [
      {
        "protocol": 6,
        "name": "tcp/139(netbios-ssn)",
        "port": 139
      },
      {
        "protocol": 6,
        "name": "tcp/445(microsoft-ds)",
        "port": 445
      }
    ]
  },
  "id": 1
}

```

Property Name	Type	Description	Notes
<i>ApplicationLayer4</i>	<object>	Object representing a mapped application.	
<i>ApplicationLayer4.enabled</i>	<string>	If the mapped application is enabled.	
<i>ApplicationLayer4.config</i>	<object>	Object representing mapped application's configuration.	
<i>ApplicationLayer4.config.port_groups</i>	<array of <object>>	Object representing mapped application's port_groups.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ]	<object>	Object representing one mapped application's port group.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ].name	<string>	Mapped application's port group name.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ].group_id	<number>	Mapped application's port group ID.	
<i>ApplicationLayer4.config.proto_ports</i>	<array of <object>>	Object representing mapped application's proto_ports.	Optional

<i>ApplicationLayer4.config.proto_ports</i> [ApplicationProtoPort]	<object>	Object representing one mapped application's proto port.	Optional
<i>ApplicationLayer4.config.proto_ports</i> [ApplicationProtoPort].port	<number>	Mapped application's port.	Optional
<i>ApplicationLayer4.config.proto_ports</i> [ApplicationProtoPort].protocol	<number>	Mapped application's protocol.	
<i>ApplicationLayer4.config.proto_ports</i> [ApplicationProtoPort].name	<string>	Mapped application's proto port name.	Optional
<i>ApplicationLayer4.config.hosts</i>	<array of <string>>	Object representing mapped application's hosts.	Optional
<i>ApplicationLayer4.config.hosts</i> [item]	<string>	Mapped application's host.	Optional
<i>ApplicationLayer4.config.cidrs</i>	<array of <string>>	Object representing mapped application's CIDRs.	Optional
<i>ApplicationLayer4.config.cidrs</i> [item]	<string>	Mapped application's CIDR.	Optional
<i>ApplicationLayer4.config.host_groups</i>	<array of <object>>	Object representing mapped application's host_groups.	Optional
<i>ApplicationLayer4.config.host_groups</i> [ApplicationHostGroup]	<object>	Object representing one mapped application's host group.	Optional
<i>ApplicationLayer4.config.host_groups</i> [ApplicationHostGroup].host_group	<number>	Mapped application's host group.	
<i>ApplicationLayer4.config.host_groups</i> [ApplicationHostGroup].name	<string>	Mapped application's host group's name.	Optional
<i>ApplicationLayer4.config.host_groups</i> [ApplicationHostGroup].host_group_type	<number>	Mapped application's host group type.	
<i>ApplicationLayer4.id</i>	<number>	Mapped application's ID.	Optional
<i>ApplicationLayer4.name</i>	<string>	Mapped application's name.	
<i>ApplicationLayer4.override</i>	<string>	Mapped application's override.	Values: Always, Unclassified, Unknown
<i>ApplicationLayer4.app_id</i>	<number>	Application ID.	Optional
<i>ApplicationLayer4.priority</i>	<number>	Mapped application's priority.	Optional

## Response Body

On success, the server returns a response body with the following structure:

JSON



```

{
  "enabled": string,
  "config": {
    "port_groups": [
      {
        "name": string,
        "group_id": number
      }
    ],
    "proto_ports": [
      {
        "port": number,
        "protocol": number,
        "name": string
      }
    ],
    "hosts": [
      string
    ],
    "cidrs": [
      string
    ],
    "host_groups": [
      {
        "host_group": number,
        "name": string,
        "host_group_type": number
      }
    ]
  },
  "id": number,
  "name": string,
  "override": string,
  "app_id": number,
  "priority": number
}

```

Example:

```

{
  "name": "CIFS",
  "enabled": true,
  "app_id": 144,
  "priority": 1,
  "override": "Unknown",
  "config": {
    "cidrs": [
      "0.0.0.0/0"
    ],
    "proto_ports": [
      {
        "protocol": 6,
        "name": "tcp/139(netbios-ssn)",
        "port": 139
      },
      {
        "protocol": 6,
        "name": "tcp/445(microsoft-ds)",
        "port": 445
      }
    ]
  },
  "id": 1
}

```

Property Name	Type	Description	Notes
<i>ApplicationLayer4</i>	<object>	Object representing a mapped application.	
<i>ApplicationLayer4.enabled</i>	<string>	If the mapped application is enabled.	
<i>ApplicationLayer4.config</i>	<object>	Object representing mapped application's configuration.	
<i>ApplicationLayer4.config.port_groups</i>	<array of <object>>	Object representing mapped application's port_groups.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ]	<object>	Object representing one mapped application's port group.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ].name	<string>	Mapped application's port group name.	Optional
<i>ApplicationLayer4.config.port_groups</i> [ <i>ApplicationPortGroup</i> ].group_id	<number>	Mapped application's port group ID.	
<i>ApplicationLayer4.config.proto_ports</i>	<array of <object>>	Object representing mapped application's proto_ports.	Optional

<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's proto port.	Optional
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].port</code>	<code>&lt;number&gt;</code>	Mapped application's port.	Optional
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].protocol</code>	<code>&lt;number&gt;</code>	Mapped application's protocol.	
<code>ApplicationLayer4.config.proto_ports [ApplicationProtoPort].name</code>	<code>&lt;string&gt;</code>	Mapped application's proto port name.	Optional
<code>ApplicationLayer4.config.hosts</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's hosts.	Optional
<code>ApplicationLayer4.config.hosts[item]</code>	<code>&lt;string&gt;</code>	Mapped application's host.	Optional
<code>ApplicationLayer4.config.cidrs</code>	<code>&lt;array of &lt;string&gt;&gt;</code>	Object representing mapped application's CIDRs.	Optional
<code>ApplicationLayer4.config.cidrs[item]</code>	<code>&lt;string&gt;</code>	Mapped application's CIDR.	Optional
<code>ApplicationLayer4.config.host_groups</code>	<code>&lt;array of &lt;object&gt;&gt;</code>	Object representing mapped application's host_groups.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup]</code>	<code>&lt;object&gt;</code>	Object representing one mapped application's host group.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].host_group</code>	<code>&lt;number&gt;</code>	Mapped application's host group.	
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].name</code>	<code>&lt;string&gt;</code>	Mapped application's host group's name.	Optional
<code>ApplicationLayer4.config.host_groups [ApplicationHostGroup].host_group_type</code>	<code>&lt;number&gt;</code>	Mapped application's host group type.	
<code>ApplicationLayer4.id</code>	<code>&lt;number&gt;</code>	Mapped application's ID.	Optional
<code>ApplicationLayer4.name</code>	<code>&lt;string&gt;</code>	Mapped application's name.	
<code>ApplicationLayer4.override</code>	<code>&lt;string&gt;</code>	Mapped application's override.	Values: Always, Unclassified, Unknown
<code>ApplicationLayer4.app_id</code>	<code>&lt;number&gt;</code>	Application ID.	Optional
<code>ApplicationLayer4.priority</code>	<code>&lt;number&gt;</code>	Mapped application's priority.	Optional

## Applications: Delete Layer 4 application

Delete Layer 4 application.

```
DELETE https://{device}/api/profiler/1.2/applications/layer_7/{layer7_app_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Applications: Update Layer 4 application

Update Layer 4 application.

```
PUT https://{device}/api/profiler/1.2/applications/layer_7/{layer7_app_id}
```

### Authorization

This request requires authorization.

### Request Body

Provide a request body with the following structure:

JSON

```

{
  "enabled": string,
  "signatures": [
    {
      "type": string,
      "signature": string
    }
  ],
  "id": number,
  "name": string,
  "sources": [
    {
      "id": number
    }
  ]
}

```

Example:

```

{
  "sources": [
    {
      "id": 0
    },
    {
      "id": 3
    },
    {
      "id": 2
    },
    {
      "id": 100
    }
  ],
  "signatures": [
    {
      "type": "Builtin",
      "signature": "\\GET daap:\\(\\.\\|\\x0a)\\iTunes^[0-9]\\.[0-9]\\.[0-9]"
    },
    {
      "type": "Builtin",
      "signature": "\\DAAP-Server: iTunes^[0-9]\\.[0-9]\\.[0-9]"
    }
  ],
  "enabled": true,
  "name": "iTunes",
  "id": 33
}

```

Property Name	Type	Description	Notes
<i>ApplicationLayer7</i>	<object>	Object representing an unmapped application.	
<i>ApplicationLayer7.enabled</i>	<string>	If the unmapped application is enabled.	
<i>ApplicationLayer7.signatures</i>	<array of <object>>	Unmapped application's hosts.	
<i>ApplicationLayer7.signatures</i> [ApplicationSignature]	<object>	Object representing an application host.	Optional
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].type	<string>	Unmapped application's type.	Values: Builtin, URL, String, Hex_String
<i>ApplicationLayer7.signatures</i> [ApplicationSignature].signature	<string>	Unmapped application's signature.	
<i>ApplicationLayer7.id</i>	<number>	Unmapped application's ID.	Optional
<i>ApplicationLayer7.name</i>	<string>	Unmapped application's name.	
<i>ApplicationLayer7.sources</i>	<array of <object>>	Unmapped application's override.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId]	<object>	Object representing an application source ID.	Optional
<i>ApplicationLayer7.sources</i> [ApplicationSourceId].id	<number>	ID of application source's ID.	

## Response Body

On success, the server does not provide any body in the responses.

## Port\_Names: Update port names

Update system port names.

```
PUT https://{device}/api/profiler/1.2/port_names
```

## Authorization

This request requires authorization.

## Request Body

Provide a request body with the following structure:

JSON

```
[
  {
    "port": number,
    "protocol": number,
    "name": string
  }
]
```

Example:

```
[
  {
    "protocol": 6,
    "name": "smtp",
    "port": 25
  },
  {
    "protocol": 6,
    "name": "nsw-fe",
    "port": 27
  }
]
```

Property Name	Type	Description	Notes
<i>CPortNameDefs</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of port name definitions.	
<i>CPortNameDefs[CPortNameDef]</i>	<i>&lt;object&gt;</i>	Object representing port name definitions.	Optional
<i>CPortNameDefs[CPortNameDef].port</i>	<i>&lt;number&gt;</i>	Port name's port.	
<i>CPortNameDefs[CPortNameDef].protocol</i>	<i>&lt;number&gt;</i>	Port name's protocol.	
<i>CPortNameDefs[CPortNameDef].name</i>	<i>&lt;string&gt;</i>	Port name's name.	

## Response Body

On success, the server does not provide any body in the responses.

## Port\_Names: List port names

Get the system port names.

```
GET https://{device}/api/profiler/1.2/port_names?offset={number}&limit={number}
```

## Authorization

This request requires authorization.

## Parameters

Property Name	Type	Description	Notes
<i>offset</i>	<i>&lt;number&gt;</i>	Starting element number.	Optional
<i>name</i>	<i>&lt;string&gt;</i>	Filter port names by the name specified.	Optional
<i>limit</i>	<i>&lt;number&gt;</i>	Number of rows to be returned.	Optional

## Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "port": number,
    "server_port": string,
    "protocol": number,
    "avg_bytes_ps": number,
    "name": string,
    "grouped": string
  }
]
```

Example:

```
[
  {
    "protocol": 6,
    "name": "smtp",
    "grouped": false,
    "server_port": true,
    "avg_bytes_ps": 32767,
    "port": 25
  },
  {
    "protocol": 6,
    "name": "nsw-fe",
    "grouped": false,
    "server_port": false,
    "avg_bytes_ps": 32767,
    "port": 27
  }
]
```

Property Name	Type	Description	Notes
<i>CPortNames</i>	<array of <object>>	List of PortNames objects.	
<i>CPortNames</i> [CPortName]	<object>	Object representing port name information.	Optional
<i>CPortNames</i> [CPortName].port	<number>	Port name's port.	
<i>CPortNames</i> [CPortName].server_port	<string>	Defined as server port.	
<i>CPortNames</i> [CPortName].protocol	<number>	Port name's protocol.	
<i>CPortNames</i> [CPortName].avg_bytes_ps	<number>	Speed information of port name.	
<i>CPortNames</i> [CPortName].name	<string>	Port name's name.	
<i>CPortNames</i> [CPortName].grouped	<string>	Used in port groups.	

## System: Start all processes (one module)

Start all system processes on one module on Enterprise systems. The operation is asynchronous. Use "GET system/{module}/status" to poll for status. The {module} can be either the IP Address or the module name.

```
POST https://{device}/api/profiler/1.2/system/{module}/start
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

## System: Get status of all processes

Get status of all system processes. On Enterprise systems, get system process statuses on all modules.

```
GET https://{device}/api/profiler/1.2/system/status
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

#### JSON

```
[
  {
    "process_id": string,
    "process_name": string,
    "module_name": string,
    "status": string,
    "module_ipaddr": string
  }
]
```

Example:

```
[
  {
    "process_id": "25096",
    "process_name": "memmonitor",
    "status": "Running"
  },
  {
    "process_name": "healthd",
    "status": "Stopped"
  },
  {
    "process_id": "25092",
    "process_name": "diskmon",
    "status": "Running"
  },
  {
    "process_id": "25123",
    "process_name": "dispatcher",
    "status": "Running"
  },
  {
    "process_name": "analyzer",
    "status": "Stopped"
  }
]
```

Property Name	Type	Description	Notes
<i>SystemStatus</i>	<array of <object>>	SystemStatus object.	
<i>SystemStatus</i> [SystemProcess]	<object>	SystemProcess object.	Optional
<i>SystemStatus</i> [SystemProcess].process_id	<string>	Process ID.	Optional
<i>SystemStatus</i> [SystemProcess].process_name	<string>	Process name.	
<i>SystemStatus</i> [SystemProcess].module_name	<string>	Module name. Available on Enterprise systems only.	Optional
<i>SystemStatus</i> [SystemProcess].status	<string>	Process status.	Values: Running, Stopped
<i>SystemStatus</i> [SystemProcess].module_ipaddr	<string>	Module IP address. Available on Enterprise systems only.	Optional

## System: Kill all processes

Kill all system processes. The operation is asynchronous. Use "GET system/status" to poll for status. On Enterprise systems, kill system processes on all modules. Warning: this operation can result in data being corrupted.

```
POST https://{device}/api/profiler/1.2/system/kill
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

## System: Restart all processes

Restart all system processes. The operation is asynchronous. Use "GET system/status" to poll for status. On Enterprise systems, stop system processes on all modules.

```
POST https://{device}/api/profiler/1.2/system/restart
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

---

## System: Start all processes

Start all system processes. The operation is asynchronous. Use "GET system/status" to poll for status. On Enterprise systems, start system processes on all modules.

```
POST https://{device}/api/profiler/1.2/system/start
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

---

## System: Restart all processes (one module)

Restart all system processes on one module on Enterprise systems. The operation is asynchronous. Use "GET system/{module}/status" to poll for status. The {module} can be either the IP Address or the module name.

```
POST https://{device}/api/profiler/1.2/system/{module}/restart
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

---

## System: Stop all processes (one module)

Stop all system processes on one module on Enterprise systems. The operation is asynchronous. Use "GET system/{module}/status" to poll for status. The {module} can be either the IP Address or the module name.

```
POST https://{device}/api/profiler/1.2/system/{module}/stop
```

### Authorization

This request requires authorization.

### Request Body

Do not provide a request body.

### Response Body

On success, the server does not provide any body in the responses.

## System: Get status of all processes (one module)

Get status of all system processes on one module on Enterprise systems. The {module} can be either the IP Address or the module name.

```
GET https://{device}/api/profiler/1.2/system/{module}/status
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
[
  {
    "process_id": string,
    "process_name": string,
    "module_name": string,
    "status": string,
    "module_ipaddr": string
  }
]
```

Example:

```
[
  {
    "process_id": "25096",
    "process_name": "memmonitor",
    "status": "Running"
  },
  {
    "process_name": "healthd",
    "status": "Stopped"
  },
  {
    "process_id": "25092",
    "process_name": "diskmon",
    "status": "Running"
  },
  {
    "process_id": "25123",
    "process_name": "dispatcher",
    "status": "Running"
  },
  {
    "process_name": "analyzer",
    "status": "Stopped"
  }
]
```

Property Name	Type	Description	Notes
<i>SystemStatus</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	SystemStatus object.	
<i>SystemStatus[SystemProcess]</i>	<i>&lt;object&gt;</i>	SystemProcess object.	Optional
<i>SystemStatus[SystemProcess].process_id</i>	<i>&lt;string&gt;</i>	Process ID.	Optional
<i>SystemStatus[SystemProcess].process_name</i>	<i>&lt;string&gt;</i>	Process name.	
<i>SystemStatus[SystemProcess].module_name</i>	<i>&lt;string&gt;</i>	Module name. Available on Enterprise systems only.	Optional
<i>SystemStatus[SystemProcess].status</i>	<i>&lt;string&gt;</i>	Process status.	Values: Running, Stopped
<i>SystemStatus[SystemProcess].module_ipaddr</i>	<i>&lt;string&gt;</i>	Module IP address. Available on Enterprise systems only.	Optional

## System: Shutdown

Shutdown the system. The operation is asynchronous.

```
POST https://{device}/api/profiler/1.2/system/shutdown
```



## Authorization

This request requires authorization.

## Request Body

Do not provide a request body.

## Response Body

On success, the server does not provide any body in the responses.

---

## System: Reboot

Reboot the system. The operation is asynchronous.

```
POST https://{device}/api/profiler/1.2/system/reboot
```

## Authorization

This request requires authorization.

## Request Body

Do not provide a request body.

## Response Body

On success, the server does not provide any body in the responses.

---

## System: Stop all processes

Stop all system processes. The operation is asynchronous. Use "GET system/status" to poll for status. On Enterprise systems, stop system processes on all modules.

```
POST https://{device}/api/profiler/1.2/system/stop
```

## Authorization

This request requires authorization.

## Request Body

Do not provide a request body.

## Response Body

On success, the server does not provide any body in the responses.

---

## System: Kill all processes (one module)

Kill all system processes on one module on Enterprise systems. The operation is asynchronous. Use "GET system/{module}/status" to poll for status. The {module} can be either the IP Address or the module name. Warning: this operation can result in data being corrupted.

```
POST https://{device}/api/profiler/1.2/system/{module}/kill
```

## Authorization

This request requires authorization.

## Request Body

Do not provide a request body.

## Response Body

On success, the server does not provide any body in the responses.

---

## Users: List users

Get a list of user accounts.

GET https://{device}/api/profiler/1.2/users

## Authorization

This request requires authorization.

## Response Body

On success, the server returns a response body with the following structure:

### JSON

```
[
  {
    "enabled": string,
    "last_name": string,
    "id": number,
    "last_login": number,
    "authentication_type": string,
    "username": string,
    "authorization_type": string,
    "role": string,
    "first_name": string,
    "last_access": number,
    "view_packet_details": string,
    "last_authentication": number,
    "view_user_information": string,
    "login_timeout": number
  }
]
```

Example:

```
[
  {
    "username": "admin",
    "last_authentication": 1352313328,
    "first_name": "Jonh",
    "last_name": "Smith",
    "authorization_type": "Local",
    "enabled": true,
    "view_user_information": true,
    "authentication_type": "Local",
    "role": "Administrator",
    "login_timeout": 900,
    "last_login": 1352313328,
    "last_access": 1352313328,
    "id": 123
  },
  {
    "username": "admin2",
    "last_authentication": 1352313328,
    "first_name": "Mark",
    "last_name": "Greg",
    "authorization_type": "Local",
    "enabled": true,
    "view_user_information": true,
    "authentication_type": "Local",
    "role": "Administrator",
    "login_timeout": 900,
    "last_login": 1352313328,
    "last_access": 1352313328,
    "id": 124
  }
]
```

Property Name	Type	Description	Notes
<i>Users</i>	<i>&lt;array of &lt;object&gt;&gt;</i>	List of user accounts on the system.	
<i>Users[User]</i>	<i>&lt;object&gt;</i>	User account.	Optional
<i>Users[User].enabled</i>	<i>&lt;string&gt;</i>	Boolean flag indicating if the user account is enabled.	
<i>Users[User].last_name</i>	<i>&lt;string&gt;</i>	Last name of the user.	
<i>Users[User].id</i>	<i>&lt;number&gt;</i>	Numeric ID of the user that the system uses internally and in the API.	
<i>Users[User].last_login</i>	<i>&lt;number&gt;</i>	Time of last login. Unix time (epoch).	
<i>Users[User].authentication_type</i>	<i>&lt;string&gt;</i>	Type of authentication for the user, such as Local or RADIUS.	Values: Local, Remote
<i>Users[User].username</i>	<i>&lt;string&gt;</i>	User name (short name) that identifies the user to the system, such as 'admin'.	

<code>Users[User].authorization_type</code>	<code>&lt;string&gt;</code>	Type of authorization for the user, such as Local or RADIUS.	Values: Local, Remote
<code>Users[User].role</code>	<code>&lt;string&gt;</code>	Role of the user. Defines permissions.	Values: Developer, Administrator, Operator, Monitor, Event_Viewer, Dashboard_Viewer
<code>Users[User].first_name</code>	<code>&lt;string&gt;</code>	First name of the user.	
<code>Users[User].last_access</code>	<code>&lt;number&gt;</code>	Time of last access to the system. Unix time (epoch).	
<code>Users[User].view_packet_details</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if the user has access to packet data.	Optional
<code>Users[User].last_authentication</code>	<code>&lt;number&gt;</code>	Time of last authentication. Unix time (epoch).	
<code>Users[User].view_user_information</code>	<code>&lt;string&gt;</code>	Boolean flag indicating if the user has access to identity information, such as Active Directory information.	Optional
<code>Users[User].login_timeout</code>	<code>&lt;number&gt;</code>	Timeout (in seconds) during which the user cannot log in to the system because of security policies.	

## Users: Re-authenticate user

Re-authenticate user account. Requires basic authentication.

```
GET https://{device}/api/profiler/1.2/users/re_authenticate
```

### Authorization

This request requires authorization.

### Response Body

On success, the server does not provide any body in the responses.

## Users: Get user

User account by user ID.

```
GET https://{device}/api/profiler/1.2/users/{user_id}
```

### Authorization

This request requires authorization.

### Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```

{
  "enabled": string,
  "last_name": string,
  "id": number,
  "last_login": number,
  "authentication_type": string,
  "username": string,
  "authorization_type": string,
  "role": string,
  "first_name": string,
  "last_access": number,
  "view_packet_details": string,
  "last_authentication": number,
  "view_user_information": string,
  "login_timeout": number
}

```

Example:

```

{
  "username": "admin",
  "last_authentication": 1352313328,
  "first_name": "Jonh",
  "last_name": "Smith",
  "authorization_type": "Local",
  "enabled": true,
  "view_user_information": true,
  "authentication_type": "Local",
  "role": "Administrator",
  "login_timeout": 900,
  "last_login": 1352313328,
  "last_access": 1352313328,
  "id": 123
}

```

Property Name	Type	Description	Notes
<i>User</i>	<object>	User account.	
<i>User.enabled</i>	<string>	Boolean flag indicating if the user account is enabled.	
<i>User.last_name</i>	<string>	Last name of the user.	
<i>User.id</i>	<number>	Numeric ID of the user that the system uses internally and in the API.	
<i>User.last_login</i>	<number>	Time of last login. Unix time (epoch).	
<i>User.authentication_type</i>	<string>	Type of authentication for the user, such as Local or RADIUS.	Values: Local, Remote
<i>User.username</i>	<string>	User name (short name) that identifies the user to the system, such as 'admin'.	
<i>User.authorization_type</i>	<string>	Type of authorization for the user, such as Local or RADIUS.	Values: Local, Remote
<i>User.role</i>	<string>	Role of the user. Defines permissions.	Values: Developer, Administrator, Operator, Monitor, Event_Viewer, Dashboard_Viewer
<i>User.first_name</i>	<string>	First name of the user.	
<i>User.last_access</i>	<number>	Time of last access to the system. Unix time (epoch).	
<i>User.view_packet_details</i>	<string>	Boolean flag indicating if the user has access to packet data.	Optional
<i>User.last_authentication</i>	<number>	Time of last authentication. Unix time (epoch).	
<i>User.view_user_information</i>	<string>	Boolean flag indicating if the user has access to identity information, such as Active Directory information.	Optional
<i>User.login_timeout</i>	<number>	Timeout (in seconds) during which the user cannot log in to the system because of security policies.	

## Users: Test RADIUS user

Test a RADIUS user.

```
POST https://{device}/api/profiler/1.2/users/radius/test_user?password={string}&username={string}
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
---------------	------	-------------	-------

<i>password</i>	<i>&lt;string&gt;</i>	RADIUS password.	
<i>username</i>	<i>&lt;string&gt;</i>	RADIUS username.	

## Request Body

Do not provide a request body.

## Response Body

On success, the server returns a response body with the following structure:

```
JSON
```

```

{
  "role_id": number,
  "error_message": string,
  "permission": string,
  "server_type": number,
  "role": string,
  "details": string,
  "permission_id": string,
  "server_ip": string,
  "authenticated": string,
  "attributes": [
    {
      [prop]: string
    }
  ],
  "authorized": string
}

```

Example:

```

{
  "error_message": "",
  "authenticated": true,
  "server_type": 2,
  "permission_id": "",
  "permission": "",
  "role_id": 0,
  "role": "",
  "authorized": false,
  "server_ip": "10.38.8.112:1812",
  "attributes": [
    {
      "25": "operatorClass"
    },
    {
      "25": "monitorClass"
    },
    {
      "25": "eventviewerClass"
    },
    {
      "17164": "unMappedRole"
    },
    {
      "17164": "monitorCascade"
    },
    {
      "17164": "eventviewerCascade"
    },
    {
      "17164": "dashboardCascade"
    },
    {
      "25": "DBAccess"
    },
    {
      "25": "dashboardClass"
    },
    {
      "17164": "AbC10~!@#%$%^&*()_+{|[];<>?/.'z"
    },
    {
      "17164": "operatorCascade"
    },
    {
      "LOGIN_SERVER": "10.38.8.112:1812"
    },
    {
      "25": "adminClass1"
    },
    {
      "25": "unMappedClass"
    },
    {
      "25": "eventviewerClass"
    },
    {
      "17164": "adminCascade"
    }
  ],
  "details": "Using 10.38.8.112:1812 - Unable to match a role."
}

```

Property Name	Type	Description	Notes
---------------	------	-------------	-------

<i>RemoteTestUserResponse</i>	<object>	RemoteTestUserResponse object.	
<i>RemoteTestUserResponse.role_id</i>	<number>	Matched role ID.	
<i>RemoteTestUserResponse.error_message</i>	<string>	Error message.	
<i>RemoteTestUserResponse.permission</i>	<string>	Matched permission name.	
<i>RemoteTestUserResponse.server_type</i>	<number>	Indicates the type of the server being tested: RADIUS(2) or TACACS+(3).	
<i>RemoteTestUserResponse.role</i>	<string>	Matched role name.	
<i>RemoteTestUserResponse.details</i>	<string>	Remote user test details.	
<i>RemoteTestUserResponse.permission_id</i>	<string>	Matched permission ID.	
<i>RemoteTestUserResponse.server_ip</i>	<string>	Remote Server IP address.	
<i>RemoteTestUserResponse.authenticated</i>	<string>	Flag indicating if the remote user was authenticated.	
<i>RemoteTestUserResponse.attributes</i>	<array of <object>>	Attributes of Remote Test User Response.	Optional
<i>RemoteTestUserResponse.attributes</i> [RemoteAttributes]	<object>	Remote attribute.	Optional
<i>RemoteTestUserResponse.attributes</i> [RemoteAttributes][prop]	<string>	Remote attribute value.	Optional
<i>RemoteTestUserResponse.authorized</i>	<string>	Flag indicating if the remote user was authorized (as Administrator, Monitor, etc).	

## Users: Test RADIUS server

Test the connection to a RADIUS server.

```
GET https://{device}/api/profiler/1.2/users/radius/test_server
```

### Authorization

This request requires authorization.

### Parameters

Property Name	Type	Description	Notes
<i>server</i>	<string>	RADIUS server identifier, example server=IP:PORT.	

### Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
  "success": string,
  "message": string
}
```

Example:

```
{
  "message": "Connection attempt succeeded",
  "success": true
}
```

Property Name	Type	Description	Notes
<i>RemoteTestServerResponse</i>	<object>	RemoteTestServerResponse object.	
<i>RemoteTestServerResponse.success</i>	<string>	Flag indicating if the remote server test was successful.	
<i>RemoteTestServerResponse.message</i>	<string>	Response message.	

## Error Codes

In the event that an error occurs while processing a request, the server will respond with appropriate HTTP status code and additional information in the response body:

```
{
  "error_id": "{error identifier}",
  "error_text": "{error description}",
  "error_info": {error specific data structure, optional}
}
```

The table below lists the possible errors and the associated HTTP status codes that may returned.

<b>Error ID</b>	<b>HTTP Status</b>	<b>Comments</b>
INTERNAL_ERROR	500	Internal server error.
AUTH_REQUIRED	401	The requested resource requires authentication.
AUTH_INVALID_CREDENTIALS	401	Invalid username and/or password.
AUTH_INVALID_SESSION	401	Session ID is invalid.
AUTH_EXPIRED_PASSWORD	403	The password must be changed. Access only to password change resources.
AUTH_DISABLED_ACCOUNT	403	Account is either temporarily or permanently disabled.
AUTH_FORBIDDEN	403	User is not authorized to access the requested resource.
AUTH_INVALID_TOKEN	401	OAuth access token is invalid.
AUTH_EXPIRED_TOKEN	401	OAuth access token is expired.
AUTH_INVALID_CODE	401	OAuth access code is invalid.
AUTH_EXPIRED_CODE	401	OAuth access code is expired.
RESOURCE_NOT_FOUND	404	Requested resource was not found.
HTTP_INVALID_METHOD	405	Requested method is not available for this resource.
HTTP_INVALID_HEADER	400	An HTTP header was malformed.
REQUEST_INVALID_INPUT	400	Malformed input structure.
URI_INVALID_PARAMETER	400	URI parameter is not supported or malformed.
URI_MISSING_PARAMETER	400	Missing required parameter.