

Policy management API v3.0

Copyright © Riverbed Technology Inc. 2024

Created Jan 16, 2024 at 02:01 PM

Resource: alert_cache

Allows generated alerts to be stored for later retrieval. The cache holds 128 alerts per policy id. Alerts whose end_time is more than an hour old will be pruned from the cache periodically. A sample algorithm to retrieve the alerts every 5 seconds: (1) query_ts = null (2) if query_ts is null, alerts = GET /alert_cache (3) else alerts = GET /alert_cache?start_time=<query_ts> (4) if alerts are empty, wait 5 seconds, go to step (2) (5) find the largest violation timestamp in the returned alerts, assign to query_ts (6) process returned alerts (7) wait 5 seconds, go to step 2

http://{device}/api/npm.policies/3.0/alert_cache

JSON

```
[
  {
    "disambiguator_list": [
      string
    ],
    "id": integer,
    "max_severity": integer,
    "notification_prefix": string,
    "overwrite_violations_on_update": boolean,
    "policy_description": string,
    "policy_id": integer,
    "policy_name": string,
    "policy_type": integer,
    "severity": integer,
    "time_range": {
      "end": string,
      "start": string
    },
    "violations": [
      {
        "attributes": VAttribute,
        "keys": VKey
      }
    ]
  }
]
```

Property Name	Type	Description	Notes
alert_cache	<array of <object>>	Allows generated alerts to be stored for later retrieval. The cache holds 128 alerts per policy id. Alerts whose end_time is more than an hour old will be pruned from the cache periodically. A sample algorithm to retrieve the alerts every 5 seconds: (1) query_ts = null (2) if query_ts is null, alerts = GET /alert_cache (3) else alerts = GET /alert_cache?start_time=<query_ts> (4) if alerts are empty, wait 5 seconds, go to step (2) (5) find the largest violation timestamp in the returned alerts, assign to query_ts (6) process returned alerts (7) wait 5 seconds, go to step 2	
alert_cache[items]	<object>	An object describing an alert on the system	Required properties: [policy_type, policy, time_range, severity];
alert_cache[items].disambiguator_list	<array of <string>>	An optional array whose members are used as a key when deciding which alerts can be merged with this one	Optional;
alert_cache[items].disambiguator_list[items]	<string>		
alert_cache[items].id	<integer>	The unique identifier of this alert	Optional; Minimum 0;
alert_cache[items].max_severity	<integer>	The maximum severity attained by this Alert over its duration	Optional; Range: 0 to 100;
alert_cache[items].overwrite_violations_on_update	<boolean>	If true, updates to this alert will overwrite previous information; if false, the update is merged into the alert	Optional;
alert_cache[items].policy_description	<string>	The description of the policy that generated this alert	Optional;
alert_cache[items].policy_id	<integer>	The unique identifier of the policy that generated this alert	Optional; Minimum 0;
alert_cache[items].policy_name	<string>	The name of the policy that generated this alert	Optional;
alert_cache[items].policy_type	<integer>	The type of the policy that generated this alert	
alert_cache[items].severity	<integer>	The current severity of this Alert	Range: 0 to 100;
alert_cache[items].time_range	<object>	A time range pair, consisting of a start and end time.	Required properties: [start, end];
alert_cache[items].time_range.end	<string>	The ending time. This is a high-precision time value in Unix Epoch format.	

<code>alert_cache[items].time_range.start</code>	<code><string></code>	The start time. This is a high-precision time value in Unix Epoch format.	
<code>alert_cache[items].violations</code>	<code><array of <object>></code>		Optional;
<code>alert_cache[items].violations[items]</code>	<code><object></code>	violating entity and value(s)	
<code>alert_cache[items].violations[items].attributes</code>	<code><VAttribute></code>	Information about the violating metric values.	
<code>alert_cache[items].violations[items].keys</code>	<code><VKey></code>		
<code>alert_cache[items].notification_prefix</code>	<code><string></code>	Used to help route notifications for this alert to the correct recipients	

Links

alert_cache: create

POST `http://{device}/api/npm.policies/3.0/alert_cache`

Request Body

Provide an [alert_cache](#) data object.

Response Body

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

alert_cache: get

GET `http://{device}/api/npm.policies/3.0/alert_cache{?policy_id,start_time,end_time}`

Response Body

Returns an [alert_cache](#) data object.

Type: PolicyCounts

JSON

```
[
  {
    "count": integer,
    "type": integer
  }
]
```

Property Name	Type	Description	Notes
<code>PolicyCounts</code>	<code><array of <object>></code>		
<code>PolicyCounts[items]</code>	<code><object></code>		
<code>PolicyCounts[items].count</code>	<code><integer></code>		Optional;
<code>PolicyCounts[items].type</code>	<code><integer></code>		Optional;

Type: SubscriptionList

JSON

```
[
  {
    "email_address": string,
    "notification_frequency": string,
    "recipient_id": integer
  }
]
```

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

<i>SubscriptionList</i>	<array of <object>>		
<i>SubscriptionList</i> [items]	<object>	Pairs a recipient ID with a notification frequency	
<i>SubscriptionList</i> [items].email_address	<string>		Optional;
<i>SubscriptionList</i> [items].recipient_id	<integer>		Optional;
<i>SubscriptionList</i> [items].notification_frequency	<string>		Values: continuous, on_crossing, on_change, on_expire;

Type: VAttribute

Information about the violating metric values.

JSON

```
{
  "additional_info_values": [
    {
      "key": string,
      "value": string
    }
  ],
  "count": integer,
  "metric_context": [
    {
      "data": [ VProfilePoint ],
      "metric": string
    }
  ],
  "metric_values": [ VValue ],
  "severity": integer,
  "timestamp": string
}
```

Property Name	Type	Description	Notes
<i>VAttribute</i>	<object>	Information about the violating metric values.	
<i>VAttribute</i> .additional_info_values	<array of <object>>	Optional information used to help display the violator	Optional;
<i>VAttribute</i> .additional_info_values[items]	<object>		
<i>VAttribute</i> .additional_info_values[items].key	<string>		Optional;
<i>VAttribute</i> .additional_info_values[items].value	<string>		Optional;
<i>VAttribute</i> .count	<integer>	The number of times this violator has appeared in the alert	Optional;
<i>VAttribute</i> .metric_context	<array of <object>>		Optional;
<i>VAttribute</i> .metric_context[items]	<object>		Required properties: [metric];
<i>VAttribute</i> .metric_context[items].data	<array of <VProfilePoint>>		Optional;
<i>VAttribute</i> .metric_context[items].data[items]	<VProfilePoint>	Represents the tolerance bands used in Adaptive Threshold alert detail reports.	
<i>VAttribute</i> .metric_context[items].metric	<string>		
<i>VAttribute</i> .metric_values	<array of <VValue>>		Optional;
<i>VAttribute</i> .metric_values[items]	<VValue>	Holds a metric and its value. Value can be null. Valid is deprecated.	
<i>VAttribute</i> .severity	<integer>	The current severity of this Violation in this Alert	Optional; Range: 0 to 100;
<i>VAttribute</i> .timestamp	<string>	Epoch timestamp representing the last time this violator was updated in the alert	Optional;

Type: VKey

JSON

```
[
  string
]
```

Property Name	Type	Description	Notes
VKey	<array of <string>>		
VKey[items]	<string>		

Type: VProfilePoint

Represents the tolerance bands used in Adaptive Threshold alert detail reports.

JSON

```
{
  "fc": number,
  "mjh": number,
  "mjl": number,
  "mnh": number,
  "mnl": number,
  "nmh": number,
  "nml": number,
  "ts": string,
  "va": number
}
```

Property Name	Type	Description	Notes
VProfilePoint	<object>	Represents the tolerance bands used in Adaptive Threshold alert detail reports.	
VProfilePoint.fc	<number>		Optional;
VProfilePoint.mjh	<number>		Optional;
VProfilePoint.mjl	<number>		Optional;
VProfilePoint.mnh	<number>		Optional;
VProfilePoint.mnl	<number>		Optional;
VProfilePoint.nmh	<number>		Optional;
VProfilePoint.nml	<number>		Optional;
VProfilePoint.ts	<string>		Optional;
VProfilePoint.va	<number>		Optional;

Type: VValue

Holds a metric and its value. Value can be null. Valid is deprecated.

JSON

```
{
  "metric": string,
  "valid": boolean,
  "value": number
}
```

Property Name	Type	Description	Notes
VValue	<object>	Holds a metric and its value. Value can be null. Valid is deprecated.	Required properties: [metric];
VValue.metric	<string>		
VValue.valid	<boolean>		Optional;
VValue.value	<number>		Optional;

Type: column_list

Collection of data column IDs.

JSON

```
[
  string
]
```

Property Name	Type	Description	Notes
<i>column_list</i>	<array of <string>>	Collection of data column IDs.	
<i>column_list</i> {items}	<string>	Data column ID.	

Type: data_def_source

Data source specification for a data definition.

JSON

```
{
  "name": string,
  "origin": string,
  "path": string,
  "reference_id": string,
  "tags": [
    string
  ],
  "type": string,
  <prop>: any
}
```

Property Name	Type	Description	Notes
<i>data_def_source</i>	<object>	Data source specification for a data definition.	Required properties: [name];
<i>data_def_source.name</i>	<string>	Name of the data source for the data definition.	
<i>data_def_source.origin</i>	<string>	The origin data source for a query that is performing a sub-query into another data source (e.g., alerting requesting into system_metrics). Required in these cases for data column validation.	Optional;
<i>data_def_source.path</i>	<string>	Reference path to the requested data from within the data source.	Optional;
<i>data_def_source.reference_id</i>	<string>	Reference to another data_def which will be used as a source to this one.	Optional;
<i>data_def_source.tags</i>	<array of <string>>	Custom labels for this data source	Optional;
<i>data_def_source.tags</i> {items}	<string>		
<i>data_def_source.type</i>	<string>	Data type requested	Optional;
<i>data_def_source.<prop></i>	<any>		Optional;