

# User filesystem API v1.0

Copyright © Riverbed Technology Inc. 2024

Created Jan 16, 2024 at 02:01 PM

## Resource: access\_rights

Access rights associated to a file system resource.

http://{device}/api/npm.filesystem/1.0/access\_rights/{+id}

### JSON

```
{
  "effective": {
    "access_right": string,
    <prop>: any
  },
  "id": string,
  "owner": string,
  "public": {
    "access_right": string,
    <prop>: any
  },
  "shared": {
    "read_only": roles_obj,
    "read_write": roles_obj,
    <prop>: any
  }
}
```

| Property Name                        | Type        | Description   | Notes   |
|--------------------------------------|-------------|---|---|
| access_rights                        | <object>    | Access rights associated to a file system resource.   |   |
| access_rights.effective              | <object>    | Provides a summary of user's access to the object. This is a read-only type.  | Required properties: [access_right]; Optional;  |
| access_rights.effective.access_right | <string>    | Access right type.  | Read-only; Values: NONE, READ_ONLY, READ_WRITE; |
| access_rights.effective.<prop>       | <any>       |   | Optional;                                       |
| access_rights.id                     | <string>    | Resource unique identifier.   | Optional;                                       |
| access_rights.owner                  | <string>    | Owner of the object.  | Read-only; Optional;                            |
| access_rights.public                 | <object>    | Indicates if an object/resource is public. All public objects/resources are read_only.  | Required properties: [access_right]; Optional;  |
| access_rights.public.access_right    | <string>    | Access right type.  | Values: READ_ONLY;                              |
| access_rights.public.<prop>          | <any>       |   | Optional;                                       |
| access_rights.shared                 | <object>    | Property read_only - Specifies a list of roles that have read_only access to the resource/object. Property read_write - Specifies a list of roles that have read_write access to the resource/object. | Optional;                                       |
| access_rights.shared.read_only       | <roles_obj> | Collections of roles object.  |   |
| access_rights.shared.read_write      | <roles_obj> | Collections of roles object.  |   |
| access_rights.shared.<prop>          | <any>       |   | Optional;                                       |

## Links

### access\_rights: get

GET http://{device}/api/npm.filesystem/1.0/access\_rights/{+id}

#### Response Body

Returns an [access\\_rights](#) data object.

### access\_rights: set

PUT http://{device}/api/npm.filesystem/1.0/access\_rights/{+id}

#### Request Body

Provide an [access\\_rights](#) data object.

#### Response Body

Returns an [access\\_rights](#) data object.

---

## Resource: checksum

Status of a checksum computation on a trace file.

```
http://{device}/api/npm.filesystem/1.0/checksum/{+id}
```

### JSON

```
{
  "id": string,
  "method": string,
  "value": string
}
```

| Property Name          | Type                  | Description                                       | Notes                                     |
|------------------------|-----------------------|---|---|
| <i>checksum</i>        | <i>&lt;object&gt;</i> | Status of a checksum computation on a trace file. | Required properties: [id, value, method]; |
| <i>checksum.id</i>     | <i>&lt;string&gt;</i> | Resource unique identifier.                       |   |
| <i>checksum.method</i> | <i>&lt;string&gt;</i> | Checksum algorithm used.                          | Values: SHA256;                           |
| <i>checksum.value</i>  | <i>&lt;string&gt;</i> | Trace file's checksum.                            |   |

---

## Links

### checksum: create

```
POST http://{device}/api/npm.filesystem/1.0/checksum/{+id}
```

#### Request Body

Do not provide a request body.

#### Response Body

Returns an [async\\_task\\_status](#) data object.

### checksum: delete

```
DELETE http://{device}/api/npm.filesystem/1.0/checksum/{+id}
```

#### Response Body

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

### checksum: get

```
GET http://{device}/api/npm.filesystem/1.0/checksum/{+id}
```

#### Response Body

Returns a [checksum](#) data object.

### checksum: status

Returns the status of the checksum computation task.

```
GET http://{device}/api/npm.filesystem/1.0/checksum_status/{+id}
```

#### Response Body

Returns an [async\\_task\\_status](#) data object.

---

## Resource: file

File or directory object.

```
http://{device}/api/npm.filesystem/1.0/fs/{+id}
```

## JSON

```
{
  "dir": file_system_dir,
  "file": virtualfile,
  "id": string,
  "type": string
}
```

| Property Name    | Type                           | Description                              | Notes                            |
|------------------|--------------------------------|--|----------------------------------|
| <i>file</i>      | <i>&lt;object&gt;</i>          | File or directory object.                | Required properties: [id, type]; |
| <i>file.dir</i>  | <i>&lt;file_system_dir&gt;</i> | Details about a directory.               |                                  |
| <i>file.file</i> | <i>&lt;virtualfile&gt;</i>     | Details about a trace or virtual file.   |                                  |
| <i>file.id</i>   | <i>&lt;string&gt;</i>          | Unique path/name for the file/directory. |                                  |
| <i>file.type</i> | <i>&lt;string&gt;</i>          | Type discriminator for filesystem items. | Values: FILE, DIRECTORY;         |

## Links

### file: copy

Copies a file or a directory to the destination path.

```
POST http://{device}/api/npm.filesystem/1.0/fileop/copy/{+id}
```

#### Request Body

Provide a request body with the following structure:

## JSON

```
{
  "destination": string
}
```

| Property Name                              | Type                  | Description                                     | Notes                               |
|--|-----------------------|---|-------------------------------------|
| <i>file.links.copy.request</i>             | <i>&lt;object&gt;</i> | Information required to copy a filesystem item. | Required properties: [destination]; |
| <i>file.links.copy.request.destination</i> | <i>&lt;string&gt;</i> | Resource's destination id.                      |                                     |

#### Response Body

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

### file: create

```
POST http://{device}/api/npm.filesystem/1.0/fs/{+id}
```

#### Request Body

Do not provide a request body.

#### Response Body

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

### file: delete

```
DELETE http://{device}/api/npm.filesystem/1.0/fs/{+id}
```

#### Response Body

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

### file: download

Downloads a file object.

```
GET http://{device}/api/npm.filesystem/1.0/fileop/download/{+id}
```

## Response Body

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

## file: get

```
GET http://{device}/api/npm.filesystem/1.0/fs/{+id}{?access,access_rights,owner}
```

## Response Body

Returns a [file](#) data object.

## file: move

Moves a file or a directory to the destination path.

```
POST http://{device}/api/npm.filesystem/1.0/fileop/move/{+id}
```

## Request Body

Provide a request body with the following structure:

### JSON

```
{
  "destination": string
}
```

| Property Name                       | Type     | Description                                     | Notes                               |
|-------------------------------------|----------|---|-------------------------------------|
| file.links.move.request             | <object> | Information required to move a filesystem item. | Required properties: [destination]; |
| file.links.move.request.destination | <string> | Resource's destination id.                      |                                     |

## Response Body

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

## Resource: filesystem

Collection of directories on the device.

```
http://{device}/api/npm.filesystem/1.0/fs
```

### JSON

```
{
  "items": [ file_system_dir ]
}
```

| Property Name           | Type                            | Description                              | Notes     |
|-------------------------|---------------------------------|--|-----------|
| filesystem              | <object>                        | Collection of directories on the device. |           |
| filesystem.items        | <array of<br><file_system_dir>> | Array of directories.                    | Optional; |
| filesystem.items[items] | <file_system_dir>               | Details about a directory.               |           |

## Links

### filesystem: get

```
GET http://{device}/api/npm.filesystem/1.0/fs{?access,access_rights,owner}
```

## Response Body

Returns a [filesystem](#) data object.

## Resource: index

Status of a microflow index on a trace file.

http://{device}/api/npm.filesystem/1.0/index/{+id}

## JSON

```
{
  "error": async_error,
  "id": string,
  "size": integer,
  "stats": {
    "items": [ index_info_stat ]
  },
  "status": string
}
```

| Property Name                   | Type                                     | Description                                  | Notes                                     |
|---------------------------------|--|--|---|
| <i>index</i>                    | <object>                                 | Status of a microflow index on a trace file. | Required properties: [id, status, stats]; |
| <i>index.error</i>              | < <i>async_error</i> >                   | Asynchronous task error object.              |   |
| <i>index.id</i>                 | <string>                                 | Index unique identifier.                     |   |
| <i>index.size</i>               | <integer>                                | Size of index on the OS file system.         | Optional;                                 |
| <i>index.stats</i>              | <object>                                 | Collection of microflow index statistics.    |   |
| <i>index.stats.items</i>        | <array of<br>< <i>index_info_stat</i> >> | Array of microflow index statistics.         | Optional;                                 |
| <i>index.stats.items[items]</i> | < <i>index_info_stat</i> >               | Microflow index statistics.                  |   |
| <i>index.status</i>             | <string>                                 | Running state of an index.                   | Values: DONE, RUNNING, STOPPED, ERROR;    |

## Links

### index: create

POST http://{device}/api/npm.filesystem/1.0/index/{+id}

#### Request Body

Do not provide a request body.

#### Response Body

Returns an [async\\_task\\_status](#) data object.

### index: delete

DELETE http://{device}/api/npm.filesystem/1.0/index/{+id}

#### Response Body

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

### index: get

GET http://{device}/api/npm.filesystem/1.0/index/{+id}

#### Response Body

Returns an [index](#) data object.

### index: status

Returns the status of the indexing creation task.

GET http://{device}/api/npm.filesystem/1.0/index\_status/{+id}

#### Response Body

Returns an [async\\_task\\_status](#) data object.

## Resource: timeskew

Status of a timeskew estimate on a multi-segment virtual file.

```
http://{device}/api/npm.filesystem/1.0/timeskew/{+id}
```

#### JSON

```
{
  "id": string,
  "results": {
    "items": [ ts_linked_file ]
  }
}
```

| Property Name                        | Type                           | Description  | Notes                               |
|--------------------------------------|--------------------------------|--|-------------------------------------|
| <i>timeskew</i>                      | <object>                       | Status of a timeskew estimate on a multi-segment virtual file. | Required properties: [id, results]; |
| <i>timeskew.id</i>                   | <string>                       | Resource unique identifier.                                    |                                     |
| <i>timeskew.results</i>              | <object>                       | Collection of linked sources details.                          |                                     |
| <i>timeskew.results.items</i>        | <array of<br><ts_linked_file>> | Array of linked sources.                                       | Optional;                           |
| <i>timeskew.results.items[items]</i> | <ts_linked_file>               | Time skew information for linked source.                       |                                     |

## Links

### timeskew: create

```
POST http://{device}/api/npm.filesystem/1.0/timeskew/{+id}{?packet_count}
```

#### Request Body

Do not provide a request body.

#### Response Body

Returns an [async\\_task\\_status](#) data object.

### timeskew: delete

```
DELETE http://{device}/api/npm.filesystem/1.0/timeskew/{+id}
```

#### Response Body

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

### timeskew: get

```
GET http://{device}/api/npm.filesystem/1.0/timeskew/{+id}
```

#### Response Body

Returns a [timeskew](#) data object.

### timeskew: status

Returns the status of the timeskew estimate task.

```
GET http://{device}/api/npm.filesystem/1.0/timeskew_status/{+id}
```

#### Response Body

Returns an [async\\_task\\_status](#) data object.

## Resource: virtualfile

Details about a trace or virtual file.

```
http://{device}/api/npm.filesystem/1.0/virtualfile/{+id}
```

## JSON

```
{
  "access_rights": access\_rights,
  "created": string,
  "format": string,
  "id": string,
  "index": index,
  "link_type": string,
  "linked_sources": file\_system\_linked\_files,
  "modified": string,
  "size": integer,
  "type": string
}
```

| Property Name                     | Type   | Description  | Notes  |
|-----------------------------------|--|--|--|
| <i>virtualfile</i>                | <object>                                     | Details about a trace or virtual file.                       | Required properties: [type, id];   |
| <i>virtualfile.access_rights</i>  | < <a href="#">access_rights</a> >            | Access rights associated to a file system resource.          |  |
| <i>virtualfile.created</i>        | < <i>string</i> >                            | Creation time in Unix time format.                           | Optional;  |
| <i>virtualfile.format</i>         | < <i>string</i> >                            | File time format.  | Optional; Values: PCAP_US, PCAP_NS, PCAPNG_US, PCAPNG_NS, UNKNOWN;   |
| <i>virtualfile.id</i>             | < <i>string</i> >                            | File unique identifier (path-like string).                   |  |
| <i>virtualfile.index</i>          | < <a href="#">index</a> >                    | Status of a microflow index on a trace file.                 |  |
| <i>virtualfile.link_type</i>      | < <i>string</i> >                            | Link layer type.   | Optional; Values: NONE, EN10MB, JUNIPER_ETHER, RAW, IEEE802_11_RADIO, PPI, PPP_WITH_DIR, LINUX_SLL, C_HDLC, UNKNOWN; |
| <i>virtualfile.linked_sources</i> | < <a href="#">file_system_linked_files</a> > | Collection of linked sources associated to the virtual file. |  |
| <i>virtualfile.modified</i>       | < <i>string</i> >                            | Modification time in Unix time format.                       | Optional;  |
| <i>virtualfile.size</i>           | < <i>integer</i> >                           | File size in bytes.  | Optional;  |
| <i>virtualfile.type</i>           | < <i>string</i> >                            | File type.   | Values: MULTISEGMENT_FILE, MERGED_FILE, PCAP_FILE, PCAPNG_FILE, ERF_FILE, UNKNOWN_FILE;                              |

## Links

### virtualfile: create

```
POST http://{device}/api/npm.filesystem/1.0/virtualfile/{+id}
```

#### Request Body

Provide a [file\\_system\\_virtual\\_file\\_import](#) data object.

#### Response Body

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

### virtualfile: set

```
PUT http://{device}/api/npm.filesystem/1.0/virtualfile/{+id}
```

#### Request Body

Provide a [file\\_system\\_virtual\\_file\\_import](#) data object.

#### Response Body

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

## Type: `async_error`

Asynchronous task error object.

## JSON



```

{
  "error_id": string,
  "error_metrics": [
    {
      "key": string,
      "value": string
    }
  ],
  "error_text": string
}

```

| Property Name                                 | Type                   | Description   | Notes  |
|---|------------------------|---|--|
| <i>async_error</i>                            | <object>               | Asynchronous task error object.   | Required properties: [error_id, error_text]; |
| <i>async_error.error_id</i>                   | <string>               | Error code.   |  |
| <i>async_error.error_metrics</i>              | <array of<br><object>> | Array of key/value error metrics, used for detailed error responses from async tasks. | Optional;                                    |
| <i>async_error.error_metrics[items]</i>       | <object>               | Individual error key/value.   | Required properties: [key, value];           |
| <i>async_error.error_metrics[items].key</i>   | <string>               | Error metric ID.  |  |
| <i>async_error.error_metrics[items].value</i> | <string>               | Error metric value.   |  |
| <i>async_error.error_text</i>                 | <string>               | Error text.   |  |

## Type: **async\_task\_status**

The status of an asynchronous task started via API calls.

### JSON

```

{
  "elapsed_time": integer,
  "error": async_error,
  "message": string,
  "progress": integer,
  "resource_uri": string,
  "status": string,
  "uri": string
}

```

| Property Name                         | Type                   | Description   | Notes                                 |
|---------------------------------------|------------------------|---|---------------------------------------|
| <i>async_task_status</i>              | <object>               | The status of an asynchronous task started via API calls.   | Required properties: [uri, progress]; |
| <i>async_task_status.elapsed_time</i> | <integer>              | Duration of the task in microseconds.   | Optional;                             |
| <i>async_task_status.error</i>        | < <i>async_error</i> > | Asynchronous task error object.   |                                       |
| <i>async_task_status.message</i>      | <string>               | Message suitable for reporting completion status to a human user, returned only when <progress> returns 100.  | Optional;                             |
| <i>async_task_status.progress</i>     | <integer>              | Integer percent completed indicator, which MUST return 100 *only* when the operation has been completed (either successfully or unsuccessfully).  | Range: 0 to 100;                      |
| <i>async_task_status.resource_uri</i> | <string>               | a URI upon which the client may perform GET operations to get the final representation of the resource that triggered the asynchronous job.   | Optional;                             |
| <i>async_task_status.status</i>       | <string>               | Asynchronous task status code.  | Optional; Values: SUCCESS, FAILURE;   |
| <i>async_task_status.uri</i>          | <string>               | URI upon which the client may perform GET operations to poll for completion. Each accepted asynchronous operation will receive a unique status URI, so that multiple operations may be initiated and tracked at once. |                                       |

## Type: **file\_system\_dir**

Details about a directory.

### JSON

```

{
  "created": string,
  "description": string,
  "dirs": filesystem,
  "files": {
    "items": file_system_files_list
  },
  "id": string,
  "modified": string
}

```

| Property Name                      | Type                     | Description  | Notes                                   |
|------------------------------------|--------------------------|--|---|
| <i>file_system_dir</i>             | <object>                 | Details about a directory.                           | Required properties: [id, files, dirs]; |
| <i>file_system_dir.created</i>     | <string>                 | Creation time in Unix time format.                   | Optional;                               |
| <i>file_system_dir.description</i> | <string>                 | Directory description used as an alias name.         | Optional;                               |
| <i>file_system_dir.dirs</i>        | <filesystem>             | Collection of directories on the device.             |   |
| <i>file_system_dir.files</i>       | <object>                 | Collection of trace and virtual files on the device. |   |
| <i>file_system_dir.files.items</i> | <file_system_files_list> | Array of trace and virtual files.                    |   |
| <i>file_system_dir.id</i>          | <string>                 | Directory unique identifier.                         |   |
| <i>file_system_dir.modified</i>    | <string>                 | Modification time in Unix time format.               | Optional;                               |

## Type: file\_system\_files\_list

Array of trace and virtual files.

JSON

```
[ virtualfile ]
```

| Property Name                        | Type                        | Description                            | Notes |
|--------------------------------------|-----------------------------|--|-------|
| <i>file_system_files_list</i>        | <array of<br><virtualfile>> | Array of trace and virtual files.      |       |
| <i>file_system_files_list[items]</i> | <virtualfile>               | Details about a trace or virtual file. |       |

## Type: file\_system\_linked\_files

Collection of linked sources associated to the virtual file.

JSON

```

{
  "items": [
    {
      "default_source": boolean,
      "description": string,
      "path": string,
      "timeskew": integer
    }
  ]
}

```

| Property Name   | Type                   | Description   | Notes                        |
|---|------------------------|---|------------------------------|
| <i>file_system_linked_files</i>                             | <object>               | Collection of linked sources associated to the virtual file.  |                              |
| <i>file_system_linked_files.items</i>                       | <array of<br><object>> | Array of linked sources.  | Optional;                    |
| <i>file_system_linked_files.items[items]</i>                | <object>               | Details about a linked source.  | Required properties: [path]; |
| <i>file_system_linked_files.items[items].default_source</i> | <boolean>              | The default source is the reference source in a multisegment file collection and it is used as reference for the time skew calculation. | Optional;                    |
| <i>file_system_linked_files.items[items].description</i>    | <string>               | A simple description field associated with the linked source.   | Optional;                    |
| <i>file_system_linked_files.items[items].path</i>           | <string>               | Linked source path. It could be either an absolute path (fs/ID) or a relative path starting from the aggregated file folder.            |                              |

|   |                              |  |           |
|---|------------------------------|--|-----------|
| <code>file_system_linked_files.items[items].timeskew</code> | <code>&lt;integer&gt;</code> | It is the packets timestamps offset associated with the linked source. | Optional; |
|---|------------------------------|--|-----------|

## Type: file\_system\_virtual\_file\_import

Info about a new multisegment or merged file.

JSON

```
{
  "linked_sources": file_system_linked_files,
  "type": string
}
```

| Property Name   | Type  | Description  | Notes  |
|---|---|--|--|
| <code>file_system_virtual_file_import</code>                | <code>&lt;object&gt;</code>                   | Info about a new multisegment or merged file.                | Required properties: [type, linked_sources]; |
| <code>file_system_virtual_file_import.linked_sources</code> | <code>&lt;file_system_linked_files&gt;</code> | Collection of linked sources associated to the virtual file. |  |
| <code>file_system_virtual_file_import.type</code>           | <code>&lt;string&gt;</code>                   | Linked file type.  | Values: MULTISEGMENT_FILE, MERGED_FILE;      |

## Type: index\_info\_stat

Microflow index statistics.

JSON

```
{
  "end_time": string,
  "start_time": string,
  "version": string
}
```

| Property Name                           | Type                        | Description                         | Notes   |
|---|-----------------------------|-------------------------------------|---|
| <code>index_info_stat</code>            | <code>&lt;object&gt;</code> | Microflow index statistics.         | Required properties: [version, start_time, end_time]; |
| <code>index_info_stat.end_time</code>   | <code>&lt;string&gt;</code> | End time for an indexed interval.   |   |
| <code>index_info_stat.start_time</code> | <code>&lt;string&gt;</code> | Start time for an indexed interval. |   |
| <code>index_info_stat.version</code>    | <code>&lt;string&gt;</code> | Microflow index version.            |   |

## Type: roles\_obj

Collections of roles object.

JSON

```
{
  "roles": [
    integer
  ],
  <prop>: any
}
```

| Property Name                       | Type  | Description                  | Notes                         |
|-------------------------------------|---|------------------------------|-------------------------------|
| <code>roles_obj</code>              | <code>&lt;object&gt;</code>                   | Collections of roles object. | Required properties: [roles]; |
| <code>roles_obj.roles</code>        | <code>&lt;array of &lt;integer&gt;&gt;</code> | Array of roles.              |                               |
| <code>roles_obj.roles[items]</code> | <code>&lt;integer&gt;</code>                  | Role value.                  |                               |
| <code>roles_obj.&lt;prop&gt;</code> | <code>&lt;any&gt;</code>                      |                              | Optional;                     |

## Type: ts\_linked\_file

Time skew information for linked source.

**JSON**

```
{  
  "path": string,  
  "timeskew": integer  
}
```

| Property Name                  | Type                   | Description   | Notes                                  |
|--------------------------------|------------------------|---|--|
| <i>ts_linked_file</i>          | <i>&lt;object&gt;</i>  | Time skew information for linked source.            | Required properties: [path, timeskew]; |
| <i>ts_linked_file.path</i>     | <i>&lt;string&gt;</i>  | Path of the linked source.                          |  |
| <i>ts_linked_file.timeskew</i> | <i>&lt;integer&gt;</i> | Estimated time skew compared to the default source. |  |