

Nasuni Data API

Nasuni Corporation – Boston, MA

Introduction

The Nasuni API has been available in the Nasuni Filer since September 2012 (version 4.0.1) and is in use by hundreds of mobile clients worldwide. Previously, the Nasuni API was only used internally by Nasuni. We are now exposing the Nasuni API to clients to use for their own integrations.

For all APIs listed here, the `host` is the host name or IP address of the Nasuni Filer, and the `port` is usually 443. All APIs use `https`.

You must enable Mobile Access for the volume that you want to use the Nasuni API with. On the Nasuni Filer user interface, use the CIFS Shares page.

Note: For all Nasuni APIs, the return codes listed are the most common and should have a specific action associated with them. It is possible that in rare cases these APIs will return other return codes, such as codes in the 500s for various server problems. For anything unspecified, the app should simply display a general error message and ask the user to try again. Report any return codes in the 500s to Nasuni Technical Support.

Note: Examples are given using [cURL](#). Since all the APIs use `https`, ensure that the version of `libcurl` that you use supports `https`. We suggest using the [cURL Download Wizard](#). When using cURL, it is necessary to handle certificates correctly. If the Nasuni Filer has only a self-signed certificate, you must disable certificate chain checking. See the description of the `-k` cURL option below.

Authenticate (Login) API

Authenticates a user's credentials and logs the user in. After authentication, all client threads must use the `X-Secret-Key` token returned for all subsequent API calls.

URL

```
https://<host>:<port>/mobileapi/1/auth/login
```

Method

POST

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

Return codes

200: Success.

401: Unauthorized. Authentication failure. Ensure that the username and password are correct. Verify that Mobile Access is enabled.

410: `device_id` is disabled on the Nasuni Filer for this particular device. Authorization using a different `device_id` is necessary.

Response headers

`X-Secret-Key`

In the case of a 200 success, this response header contains a secret key, which the app is responsible for saving and using on all subsequent API calls. This key can be used concurrently by any number of client threads. If the secret key is lost for any reason, you can just re-authenticate to get a new one.

This response header is of the form:

```
X-Secret-Key:  
gWTklSZs9EpuAo3yM3TymvdbzcaKeQ6H37EnLK6  
spUs=
```

cURL example

Note: In this example, the cURL switches include the following:

-k

This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.

-i

Include the HTTP-header in the output. The HTTP-header includes information like server-name, date of the document, HTTP-version and more.

-F

This lets curl emulate a filled-in form in which a user has pressed the Submit button.

Note: In this example, the following fields are used:

username

Active Directory username. Both `<username>` and `<domain>\<username>` formats are allowed.

password

Password for the username.

Note: applications should never save the user's password.

device_id

The field name "device_id" comes from the Mobile Access API, which is the origin of the Nasuni API. This field allows you to enter a unique random string, such as a GUID, that uniquely identifies the client.

Note: this device_id appears on the Mobile License Usage page of the Nasuni Filer user interface.

device_type

The field name "device_type" comes from the Mobile Access API. Select one of the following: "iphone", "ipad", "android", "windows"‡, "osx"‡, or "linux"‡.

Note: this allows you to enable or disable access by type. If you need another type, notify Nasuni Technical Support.

‡ Planned. To use, contact Nasuni Technical Support.

```
curl -k -i
-F username=TESTUSER -F password=TESTPASSWORD
-F device_id=apple001 -F device_type=iphone
https://hostname/mobileapi/1/auth/login
```

Output:

```
HTTP/1.1 100 Continue
```

```
HTTP/1.1 200 OK
```

```
Date: Tue, 01 May 2012 20:38:00 GMT
```

```
Server: Apache
```

```
X-Secret-Key:
```

```
gWTKlSZs9EpuAo3yM3TymvdbzcaKeQ6H37EnLK6spUs=
```

```
Content-Length: 0
```

```
Connection: close
```

```
Content-Type: text/plain; charset=UTF-8
```

Logout

De-authenticates the user from the interface. After use of this API, the caller should forget the secret key, but can still retain and re-use the device ID.

URL

`https://<host>:<port>/mobileapi/1/auth/logout`

Method

POST

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

Return codes

200: Success.

401: Unauthorized. Authentication failure. Authentication failures on non-authenticated APIs can be due to expired secret keys. On logout, this error can be ignored, and the logout can be considered successful.

cURL example

Note: In this example, the cURL switches include the following:

- X
Specifies a custom request method to use when communicating with the HTTP server. The specified request is used instead of the method otherwise used (which defaults to GET).
- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- i
Include the HTTP-header in the output. The HTTP-header includes information like server-name, date of the document, HTTP-version and more.
- u
Use "device_id:X-Secret-Key".

Note: In this example, the following fields are used:

- device_id
The same device_id used for login.
- X-Secret-Key
The secret key obtained after running login.

```
curl -X POST -k -i
-u "apple001:gWtk1SZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/auth/logout
```

Output:

```
HTTP/1.1 200 OK
Date: Tue, 01 May 2012 20:38:00 GMT
Server: Apache
Content-Length: 0
Connection: close
Content-Type: text/plain; charset=UTF-8
```

Get items

Get the contents of an item, either a directory listing, the previous version listing of a file or directory, or the contents of a file.

URL

`https://<host>:<port>/mobileapi/1/fs/<path>`

Note: “fs” stands for “file system”.

Method

GET

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Range: [HTTP Range header](#). Optional. Only meaningful if the path is to a file.

‡X-NAS-Options:generate_etag. Use entity tags (ETag) [ETag Specification](#) to allow client caching of results.

‡If-None-Match: ETag. ETag from previous GET request. A return code of 304 is returned if this directory or file has not changed per the ETag. See [ETag Behavior](#) below.

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

path

(Optional.) Path to the item. If omitted, the result is the top directory.

‡ Planned. To use, contact Nasuni Technical Support.

Return codes

- 200: Success.
- 206: Partial range satisfied.
- 304: Not modified.
- 401: Authentication failure.
- 403: Permission denied.
- 404: Object not found.
- 410: `device_id` is disabled on the Nasuni Filer for this particular device.
Authorization using a different `device_id` is necessary.
- 416: Range requested but cannot be satisfied.

Response headers:

- X-Object-Type: One of `shares`, `share`, `directory`, `versions`, or `file`.
- ‡ETag: If client requested generation of an ETag (see X-NAS-Options above). See [ETag Behavior](#) below.

Response body:

- If `shares`, the list of shares available to navigate into.
- If `directory` or `share`, the list of files and directories within this directory.
- If `versions`, a listing of the previous versions of a file or directory.
- If `file`, the (possibly partial) contents of the file.

‡ Planned. To use, contact Nasuni Technical Support.

Example: obtaining top-level directory

Note: In this example, the cURL switches include the following:

```
-k
    This option explicitly allows curl to perform "insecure" SSL
    connections and transfers. All SSL connections are attempted to be
    made secure by using the CA certificate bundle installed by default.
    This makes all connections considered "insecure" fail unless -k or
    --insecure is used.

-u
    Use "device_id:X-Secret-Key"
```

Note: In this example, the following fields are used:

```
device_id
    The same device_id used for login.

X-Secret-Key
    The secret key obtained after running login.
```

```
curl -k
-u "apple001:gWTKlSZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs
```

Note: the path is omitted to get the top directory.

The result, in [JSON](#) format, is of the form:

```
{
  "name": "/",
  "items": [
    {"type": "share", "name": "share1", "mod_time":
      1334857842},
    {"type": "share", "name": "share2", "mod_time":
      1334857347},
    {"type": "share", "name": "share3", "mod_time":
      1334857436}
  ]
}
```

Example: obtaining arbitrary directory

Note: In this example, the cURL switches include the following:

- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- u
Use `"device_id:X-Secret-Key"`

Note: In this example, the following fields are used:

- device_id
The same device_id used for login.
- X-Secret-Key
The secret key obtained after running login.

```
curl -k
-u "apple001:gWtk1SZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/dir
```

The result, in [JSON](#) format, is of the form[‡]:

```
{
  "name": "/full/path/to/dir",
  "items": [
    {"mod_time": 1334857842, "type": "file", "name":
      "file2", "size": 10240000, "ETag": "1-1334857842-
      10240000"},
    {"mod_time": 1234567890, "type": "file", "name":
      "file4", "size": 324, "ETag": "1-1234567890-
      324"},
    {"mod_time": 1234567890, "type": "directory",
      "name": "dir2"}
  ]
}
```

[‡] ETag support planned. To use, contact Nasuni Technical Support.

Example: obtaining previous versions of a file

Note: In this example, the cURL switches include the following:

`-k`

This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.

`-u`: Use "device_id:X-Secret-Key"

Note: In this example, the following fields are used:

`previous`

(Optional.) True or false. Instead of getting a file or directory's contents, get its list of previous versions.

`device_id`: The same `device_id` used for login.

`X-Secret-Key`: The secret key obtained after running login.

```
curl -k
-u "apple001:gWTKlSZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/file
?previous=true
```

The result, in [JSON](#) format, is of the form[‡]:

```
{
  "type": "file",
  "name": "/full/path/to/filename",
  "versions": [
    {"version_time":1336049867, "mod_time":
      1336049977, size: 84576, "ETag": "1-1336049867-
      84576"},
    {"version_time":1336049977, "mod_time":
      1336049537, size: 74537, "ETag": "1-1336049977-
      74537"},
    {"version_time":1336050987, "mod_time":
      1335996049, size: 63452, "ETag": "1-1336050987-
      63452"}
  ]
}
```

[‡] ETag support planned. To use, contact Nasuni Technical Support.

Example: obtaining previous versions of a directory

Note: In this example, the cURL switches include the following:

```
-k
    This option explicitly allows curl to perform "insecure" SSL
    connections and transfers. All SSL connections are attempted to be
    made secure by using the CA certificate bundle installed by default.
    This makes all connections considered "insecure" fail unless -k or
    --insecure is used.

-u
    Use "device_id:X-Secret-Key"
```

Note: In this example, the following fields are used:

```
previous
    (Optional.) True or false. Instead of getting a file or directory's
    contents, get its list of previous versions.

device_id
    The same device_id used for login.

X-Secret-Key
    The secret key obtained after running login.
```

```
curl -k
-u "apple001:gWTKlSZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/dir
?previous=true
```

The result, in [JSON](#) format, is of the form:

```
{
  "type": "directory",
  "name": "/full/path/to/dirname",
  "versions": [
    {"version_time":1336049867, "mod_time":
      1336049977},
    {"version_time":1336049977, "mod_time":
      1336049537},
    {"version_time":1336050987, "mod_time":
      1335986914}
  ]
}
```

Example: downloading a single file

Note: In this example, the cURL switches include the following:

- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- o <local-path-to- file>
Output file format.
- u
Use "device_id:X-Secret-Key"

Note: In this example, the following fields are used:

- device_id
The same device_id used for login.
- X-Secret-Key
The secret key obtained after running login.

```
curl -k  
-u "apple001:gWtk1SZs9EpuAo3yM3TymvdbzcaKeQ6H37="  
-o <local-path-to- file>  
https://hostname/mobileapi/1/fs/full/path/to/file
```

Notes for fs

This field is also available:

`version_time`

(Optional.) Specifies which version of the file or directory to retrieve.

It is possible to continue traversing the previous version of a directory to deeper directories and files. For example, suppose you are looking at version 1336050987 of directory /A/B/C, which contains a directory D. You can get the contents of D by issuing a GET request to /A/B/C/D?version_time=1336050987.

Furthermore, if D contains a file F, you can retrieve that file by issuing a GET request to /A/B/C/D/F?version_time=1336050987.

However, you cannot see the previous versions of F (or D). That is, this request is invalid: /A/B/C/D/F?version_time=1336050987&previous=true.

Calling GET on the root directory / results in a list of shares.

Note: There are no previous versions of /.

Upload file

Upload a file to the Nasuni Filer.

Note: you cannot upload a file to the root level, since that is a list of CIFS shares.

URL

```
https://<host>:<port>/mobileapi/1/fs/  
<path_including_new_filename>
```

Note: “fs” stands for “file system”.

Method

PUT

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Content-Length: the size of the file to upload.

Request Body

The raw bytes to be stored.

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

path_including_new_filename

Path to the new file in the Nasuni Filer, including the new filename.

Return codes

200: Success.

401: Authentication failure.

403: Permission denied.

404: Parent directory not found.

409: File or directory already exists.

410: `device_id` is disabled on the Nasuni Filer for this particular device.
Authorization using a different `device_id` is necessary.

cURL example

Note: In this example, the cURL switches include the following:

- X
Specifies a custom request method to use when communicating with the HTTP server. The specified request is used instead of the method otherwise used (which defaults to GET).
- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- u
Use "`device_id:X-Secret-Key`"
- T <file>
This transfers the specified local file to the remote URL. You can specify one `-T` for each URL on the command line. Each `-T + URL` pair specifies what to upload and to where.

Note: In this example, the following fields are used:

- `device_id`
The same `device_id` used for login.
- `X-Secret-Key`
The secret key obtained after running login.

```
curl -X PUT -k -T file.txt
-u "apple001:gWTKlSZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/
newfilename.txt
```


Notes for fs

These fields are also available:

`mod_time`

(Optional.) Integer. The last modification time of the file on upload, in seconds since 1970.

`overwrite`

(Optional.) True or false. Overwrite destination file if it exists.

`createparents`

(Optional.) True or false. Create parent directories if they do not exist.

Download files

Download multiple files as a zip archive.

URL

```
https://<host>:<port>/mobileapi/1/fs/  
<path>?action=MULTIDL
```

Note: “fs” stands for “file system”.

Method

POST

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Parameters

action

multidl

paths

Multiple entries containing the relative paths to include in the archive. Path names do not include the path provided in the URL.

See examples:

- paths = /1920x1200
- paths = /models
- paths = /1280x1024/raver_1280x1024.jpg

Return codes

200: Success.

400: Invalid input, such as a directory name that ends in a dot, which is illegal on Windows.

401: Authentication failure.

403: Permission denied.

404: Parent directory not found.

409: Conflict. Reach max-size or max-files constraint when building zip archive.

411: No paths specified.

cURL example

Note: In this example, the cURL switches include the following:

-k

This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless -k or --insecure is used.

-u

Use "device_id:X-Secret-Key"

-o <local-path-to-zip-file>

Output file in .zip format.

Note: In this example, the following fields are used:

device_id

The same device_id used for login.

X-Secret-Key

The secret key obtained after running login.

```
curl -k -u "device_id:X-Secret-Key"  
-o <local-path-to-zip-file>  
https://hostname/mobileapi/1/fs/full/path/to/filename.  
extension?action=MULTIDL
```

Create directory

Create a directory in the Nasuni Filer.

Note: you cannot create a directory at the root level, since that is a list of CIFS shares.

URL

```
https://<host>:<port>/mobileapi/1/fs/  
<path_including_dir_name>
```

Note: “fs” stands for “file system”.

Method

POST

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

path_including_dir_name

Path to the new directory in the Nasuni Filer, including the new directory name.

Return codes

200: **Success**.

204: **No content (success)**.

400: **Invalid input**, such as a directory name that ends in a dot, which is illegal on Windows.

401: **Authentication failure**.

403: **Permission denied**.

404: **Parent directory not found**.

409: **File or directory already exists**.

410: `device_id` is disabled on the Nasuni Filer for this particular device.
Authorization using a different `device_id` is necessary.

cURL example

Note: In this example, the cURL switches include the following:

- X
Specifies a custom request method to use when communicating with the HTTP server. The specified request is used instead of the method otherwise used (which defaults to GET).
- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- u
Use `"device_id:X-Secret-Key"`

Note: In this example, the following fields are used:

- `action=MKDIR`
Action to make a new directory.
- `device_id`
The same `device_id` used for login.
- `X-Secret-Key`
The secret key obtained after running login.

```
curl -X POST -k
-u "apple001:gWtklSZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/dir/
?action=MKDIR
```

Notes for fs

This field is also available:

- `mod_time`
(Optional.) Integer. The last modification time of the directory on upload, in seconds since 1970.

Delete file or directory

Delete a file or directory.

Note: you cannot delete a non-empty directory. You also cannot delete a directory at the root level, since those are technically shares, and you cannot delete a non-empty directory.

URL

```
https://<host>:<port>/mobileapi/1/fs/  
<full_path_to_file_or_dir>
```

Note: “fs” stands for “file system”.

Method

DELETE

Request Headers

Authorization: Basic <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

full_path_to_file_or_dir

Return codes

204: Deleted successfully.

401: Authentication failure, such as bad device ID, username, or password.

403: Permission denied.

404: File or directory not found.

409: Directory not empty.

410: device_id is disabled on the Nasuni Filer for this particular device.
Authorization using a different device_id is necessary.

cURL example

Note: In this example, the cURL switches include the following:

-X

Specifies a custom request method to use when communicating with the HTTP server. The specified request is used instead of the method otherwise used (which defaults to GET).

-k

This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.

-u

Use "device_id:X-Secret-Key"

Note: In this example, the following fields are used:

device_id

The same device_id used for login.

X-Secret-Key

The secret key obtained after running login.

```
curl -X DELETE -k  
-u "apple001:gWtk1SZs9EpuAo3yM3TymvdbzcaKeQ6H37="  
https://hostname/mobileapi/1/fs/full/path/to/  
file_or_dir
```

Exists query

Check if a file or directory exists.

URL

`https://<host>:<port>/mobileapi/1/fs/<path>`

Note: “fs” stands for “file system”.

Method

HEAD

Request Headers

Authorization: **Basic** <device_id>:<X-Secret-Key> - mandatory
[Basic Auth Header](#).

Response headers:

X-Size: Size of file.

X-Modtime: Modification date of file or directory.

X-Object-Type: One of shares, share, directory, versions, or file.

Parameters

host

The host name or IP address of the Nasuni Filer.

port

The port of the Nasuni Filer, usually 443. If the port is 443, you can omit the port part, unless you are using network address translation (NAT).

Return codes

200: Success.

401: Authentication failure, such as bad device ID, username, or password.

403: Permission denied.

404: File or directory not found.

410: `device_id` is disabled on the Nasuni Filer for this particular device.
Authorization using a different `device_id` is necessary.

cURL example

Note: In this example, the cURL switches include the following:

- i
Include the HTTP-header in the output. The HTTP-header includes information like server-name, date of the document, HTTP-version and more.
- X
Specifies a custom request method to use when communicating with the HTTP server. The specified request is used instead of the method otherwise used (which defaults to GET).
- k
This option explicitly allows curl to perform "insecure" SSL connections and transfers. All SSL connections are attempted to be made secure by using the CA certificate bundle installed by default. This makes all connections considered "insecure" fail unless `-k` or `--insecure` is used.
- u
Use "device_id:X-Secret-Key"

Note: In this example, the following fields are used:

- device_id
The same device_id used for login.
- X-Secret-Key
The secret key obtained after running login.

```
curl -i -X HEAD -k
-u "apple001:gWtk1SZs9EpuAo3yM3TymvdbzcaKeQ6H37="
https://hostname/mobileapi/1/fs/full/path/to/
file_or_dir
```

Output:

```
HTTP/1.1 200 OK
Date: Fri, 04 May 2012 20:22:21 GMT
Server: Apache
X-Size: 0
X-Modtime: 1336162154
X-Object-Type: directory
Connection: close
Content-Type: text/plain; charset=UTF-8
```

List of return codes

- 200: Success.
- 204: Deleted successfully.
- 206: Partial range satisfied.
- 304: Not modified.
- 400: Invalid input, such as a directory name that ends in a dot, which is illegal on Windows.
- 401: Authentication failure, such as bad device ID, username, or password.
- 403: Permission denied.
- 404: Object not found.
- 409: File or directory already exists. Or directory not empty.
- 410: `device_id` is disabled on the Nasuni Filer for this particular device. Authorization using a different `device_id` is necessary.
- 416: Range requested but cannot be satisfied.

ETag Behavior‡

The intent of ETag is to let the caller know that an item has not changed if it has the same ETag as before. This provides some efficiencies in the synchronization process.

Directories

ETag for directories is only present (not blank or empty) when the directory has been fully synchronized with the cloud.

Any changes within a directory (such as new file, file rename, or file create) immediately clears the ETag on the directory. The ETag remains missing or blank until that directory is fully synchronized with the cloud.

When encountering a directory with a missing ETag, you must enumerate or process all contents, comparing the ETag of each item within, and for directories descending, if needed.

The directory ETag is updated before file updates are done so there are no race conditions.

Files

ETag is always present and is continually updated as a file's contents are changed. This is not dependent on the snapshot cycle.

ETag is not affected by things that change metadata such as rename, permissions, and ACLs.

Note that files can have the same ETag but are not the same file. ETag doesn't include anything like a hash.

Note: file metadata operations, including renames, are really directory operations to us.

Troubleshooting

If you have any questions about original or added protocols for your volume, contact Nasuni Technical Support.

PHONE: 1.857.444.8500 / 1.800.208.3418

EMAIL: support@nasuni.com

WEB: <https://www.nasuni.com/support/>

Copyright © 2010-2018 Nasuni Corporation. All rights reserved.

‡ ETag support planned. To use, contact Nasuni Technical Support.