
Cavium-Nas Documentation

Release 0.2

Sreeni Puthiyillam

March 03, 2013

CONTENTS

Contents:

INTRODUCTION

1.1 Welcome to Cavium NAS targeted for Small Office Home Office segment.

Cavium NAS is the Cavium's solution for Network Attached Storage (NAS). Cavium NAS packs a wide set of functionalities required for a NAS device. This also offers a high degree of customization to cater design specific solution variants. The NAS middleware is tightly integrated with Cavium SoC's hardware SDK for optimal performance. The NAS middleware provides easy to use, middle-ware APIs to build a lively, custom NAS user interface.

1.2 features

- Storage (disk) management
- Logical volume management
- File sharing protocols (CIFS, FTP in this version)
- Storage access management – by configuring users and groups access
- Network configuration
- NAS UI

1.3 Availability

The Cavium NAS solution is available on Cavium' Econa® and Octeon® SoC platforms.

1.4 Contact

Cavium Network - DHOD - NAS-TEAM

STATUS AND RESPONSES

Statuses and responses are known via the following:

- [Status Codes](#)
- [Error Messages](#)

2.1 Status Codes

All APIs should return appropriate [HTTP Status codes](#).

Code	Text	Description
200	OK	Success
304	Not Modified	No new data. (Cached)
400	Bad Request	The request was invalid. The response will contain an error message explaining the same.
401	Unauthorized	Authentication credentials missing or invalid.
403	Forbidden	The user does not have access to that particular resource. The response will contain an error message explaining the same.
404	Not Found	Invalid URI or resources does not exist.
408	Request Timeout	Server was busy in processing and timed out.
500	Internal Server Error	Server is broken.

2.2 Error Messages

Error messages give more detail on the error occurred. They are of the following format:

```
{
  "errors": [
    {
      "message": "Sorry, that folder does not exist",
      "code": 234
    }
  ]
}
```


STORAGE

Note: This is a work in progress

The storage api encompasses the following sub modules.

- *Disks*
- *Volumes*
- *Shares*

3.1 iSCSI

The iscsi api exposes interfaces related to iscsi present in the the NAS device.

DISKS

The disks api exposes interfaces related to disks present in the the NAS device.

4.1 Disk Object

The following represents a disk object:

```
[
  {
    "id": "",
    "name": "",
    "description": "",
    "path": "",
    "size": "",
    "uuid": "",
    "status": "",
    "vendor": "",
    "volumes": [
      {
        "name": "",
        "id": "",
        "size": "",
        "used": "",
        "status": "",
        "encrypted": false
      }
    ],
    "temperature": "",
    "model": "",
    "serial": "",
    "smart": {
      "enabled": true,
      "status": "",
      "progress": ""
    },
    "actions": {
      "eject": true,
      "claim": true
    }
  }
]
```

Note: By default all values are strings. If the value is an array or boolean, it would be mentioned.

Disk Object Details

Name	Value
id	The primary key unique id by which the disk can be identified
name	The name of the disk
description	Description of the disk
path	The physical path of the disk in the NAS device.
size	Total size of the disk in bytes.
uuid	Unique hardware id of the disk.
status	Status of the disk. Can be one of the four values good bad foreign uninstalled
vendor	Manufacturing vendor of the disk.
	Array containing brief information of volumes created in this disk.
name	The name of the volume.
id	The primary key unique id by which volume can be identified.
size	Total size of the volume in bytes.
used	Size of the volume used in bytes.
status	Status of the volume as described in Volume Status .
encrypted	Boolean value which says whether a volume is encrypted or not.
temperature	Temperature of the disk.
model	Model of the disk.
serial	Disk manufacturer serial number.
	Self-Monitoring, Analysis and Reporting Technology System details
status	Overall S. M. A. R. T health status of the disk
smart	
enabled	Currently available status of S.M.A.R.T
Progress	Currently available progress of S.M.A.R.T
	Actions which can be done on the disk.
ejections	Boolean value which says whether ejecting this disk is possible.

claim | **Boolean value which says whether** claiming this disk is possible.

Error messages

Error messages will be of the format described in [Error Messages](#).

Code	Description
100	Operation not allowed. Used for validation, or blocking an action because another action is going on.
101	Unable to complete the action. Used for internal errors.

4.2 GET - Disks

Returns [Disk object](#) containing the disks present in the NAS device.

Resource URL -> <nas-box-ip-address>/index.php/disks/api

Input -> None

Response -> Array of [Disk objects](#)

4.3 PUT - Claim Disk

Takes [Disk object](#) containing the disk to claim. The API claims the foreign disk.

Note: Operation can be performed only on foreign disks.

Resource URL -> <nas_box_ip_address>/index.php/disks/api

Input -> [Disk object](#)

Response -> [Disk object](#) which got claimed.

4.4 POST - Eject Disk

Takes [Disk object](#) containing the disk to eject. The API ejects the disk safely.

Resource URL -> <nas_box_ip_address>/index.php/disks/api

Input -> [Disk object](#)

Response -> [Disk object](#) which got ejected.

VOLUMES

The volumes api exposes interfaces related to volumes present in the the NAS device.

5.1 Volume Object

The following represents a volume object:

```
[
  {
    "id": "",
    "name": "",
    "description": "",
    "raid": "",
    "size": "",
    "used": "",
    "status": "",
    "encrypted": false,
    "raw": false,
    "additional_info": {
      "rate_of_progress": "",
      "estimated_time": "",
      "disk_speed": ""
    },
    "disks": [
      {
        "name": "",
        "id": "",
        "size": "",
        "used": "",
        "status": ""
      }
    ],
    "shares": [
      {
        "name": "",
        "id": ""
      }
    ],
    "actions": {
      "edit": true,
      "delete": true,
      "migrate": {
```

```

    "to_raid1": false,
    "to_raid5": false,
    "to_raid10": false,
    "disks": [],
    "mode": ""
  },
  "extend": {
    "disks": [],
    "mode": ""
  },
  "recover": false
}
]

```

Note: By default all values are strings. If the value is an array or boolean, it would be mentioned.

Volume Object Details

	Name	Value
id	The primary key unique id by which the volume can be identified	
name	The name of the volume	
description	Description of the volume	
raid	Raid type of the volume	
size	Total size of the volume in bytes.	
used	Size of the volume used in bytes.	
status	Status of the volume. Can be one of the seven values good degraded failed recovering resizing building	
rate_of_progress	Additional information which is got only volume status recovering, transferring, resizing and building.	
estimated_time	Progress rate at which the action is happening.	
disk_speed	Estimated time of completion of the action.	
disks	Speed at which the disk spins.	
name	Array containing brief information of disks used for this volume.	
id	The name of the disk.	
size	The primary key unique id by which disk can be identified.	
used	Total size of the disk in bytes.	
status	Size of the disk used in bytes.	
shares	Status of the disk as described in Disk Status .	
id	Array containing brief information of shares present in this volume.	
encrypted	The name of the share.	
raw	The primary key unique id by which share can be identified.	
edit	Boolean value which says whether the volume is encrypted or not.	
delete	Boolean value which says whether the volume is raw (without filesystem) or not.	
to_raid1	Actions which can be done on the volume.	
to_raid5	Boolean value which says whether editing this volume is possible.	
to_raid10	Boolean value which says whether deleting this volume is possible.	
disks	Details of migrations possible in this volume.	
	Boolean - to raid1	
	Boolean - to raid5	
	Boolean - to raid10	
	Array of disks ids which can be used for migrating.	

Continued on next page

Table 5.1 – continued from previous page

Name		Value
mode	Mode of migrating. Can be <ul style="list-style-type: none"> • online • offline 	
disk end	Details of whether the raid can be extended with additional disks. Array of disks ids which can be used for extending.	
mode	Mode of extending. Can be <ul style="list-style-type: none"> • online • offline 	
recover	Boolean value which says whether the raid can be recovered.	

5.2 GET Volumes

Returns [Volume object](#) containing the volumes present in the NAS device.

Resource URL http://<nas_box_ip_address>/index.php/volumes/api

Input → None

Response → Array of [Volume objects](#)

5.3 POST - Create Volume

Takes [Volume object](#) containing the volume to create.

Resource URL → <nas_box_ip_address>/index.php/volumes/api

Input → [Volume object](#)

Response → [Volume object](#) which got created with a new *id*.

5.4 DELETE - Delete Volume

Takes [Volume object](#) containing the volume to delete.

Resource URL → <nas_box_ip_address>/index.php/volumes/api

Input → [Volume object](#)

Response → [Volume object](#) which got deleted.

5.5 PUT - Edit Volume

Takes [Volume object](#) containing the volume to edit.

Note: The “edit” boolean attribute in “actions” should be true.

Resource URL -> <nas_box_ip_address>/index.php/volumes/api

Input -> Volume object

Response -> Volume object which got edited.

5.6 PUT - Migrate Volume

Takes Volume object containing the volume to migrate.

Note: The “migrate” boolean attribute in “actions” should be true.

Resource URL -> <nas_box_ip_address>/index.php/volumes/api

Input -> Volume object

Response -> Volume object which got migrated.

5.7 PUT - Extend Volume

Takes Volume object containing the volume to extend.

Note: The “extend” boolean attribute in “actions” should be true.

Resource URL -> <nas_box_ip_address>/index.php/volumes/api

Input -> Volume object

Response -> Volume object which got extendd.

5.8 PUT - Recover Volume

Takes Volume object containing the volume to recover.

Note: The “recover” boolean attribute in “actions” should be true.

Resource URL -> <nas_box_ip_address>/index.php/volumes/api

Input -> Volume object

Response -> Volume object which got recovered.

SHARES

The shares api exposes interfaces related to shares present in the the NAS device.

6.1 Share Object

The following represents a share object:

```
[
  {
    "id": "",
    "name": "",
    "description": "",
    "volume": {
      "name": "",
      "id": "",
      "size": "",
      "used": "",
      "status": "",
      "encrypted": false
    },
    "public": false,
    "cifs": {
      "enabled": true,
      "readonly": {
        "users": [
          {
            "name": "",
            "id": ""
          }
        ]
      },
      "groups": [
        {
          "name": "",
          "id": ""
        }
      ]
    },
    "fullaccess": {
      "users": [
        {
          "name": "",
          "id": ""
        }
      ]
    }
  }
]
```

```
    }
  ],
  "groups": [
    {
      "name": "",
      "id": ""
    },
  ],
]
},
"noaccess": {
  "users": [
    {
      "name": "",
      "id": ""
    }
  ],
  "groups": [
    {
      "name": "",
      "id": ""
    },
  ],
]
}
},
"afp": {
  "enabled": true,
  "readonly": {
    "users": [
      {
        "name": "",
        "id": ""
      }
    ],
    "groups": [
      {
        "name": "",
        "id": ""
      },
    ],
  ],
},
"fullaccess": {
  "users": [
    {
      "name": "",
      "id": ""
    }
  ],
  "groups": [
    {
      "name": "",
      "id": ""
    },
  ],
],
},
"noaccess": {
  "users": [
    {
      "name": "",
```

```

        "id": ""
    }
],
"groups": [
    {
        "name": "",
        "id": ""
    },
]
}
},
"nfs": {
    "enabled": true,
    "ips": [
        ""
    ]
},
"ftp": {
    "enabled": true,
    "users": [
        {
            "name": "",
            "id": ""
        }
    ],
    "groups": [
        {
            "name": "",
            "id": ""
        },
    ]
},
"webdav": {
    "enabled": true,
    "readonly": {
        "users": [
            {
                "name": "",
                "id": ""
            }
        ],
        "groups": [
            {
                "name": "",
                "id": ""
            },
        ]
    },
},
"fullaccess": {
    "users": [
        {
            "name": "",
            "id": ""
        }
    ],
    "groups": [
        {
            "name": "",

```

```

        "id": ""
    },
]
},
"noaccess": {
    "users": [
        {
            "name": "",
            "id": ""
        }
    ],
    "groups": [
        {
            "name": "",
            "id": ""
        }
    ],
}
},
"recycle_bin": {
    "enabled": false
},
"media_service": {
    "enabled": false
}
}
]

```

Share Object Details

	Name Value	
id	The primary key unique id by which the share can be identified	
name	The name of the share	
description	Description of the share	
	Brief info of the volume on which the share is created.	
name	Name of the volume.	
id	Unique primary key id of the volume.	
size	Size of the volume in bytes.	
used	Size of the volume used in bytes.	
status	Status of the volume as specified in Volume Status .	
encrypted	Boolean value which says whether a volume is encrypted or not.	
public	Boolean value which tells whether the share is public or private.	
enabled	Details of CIFS/AFP/Webdav services in the share. Boolean value which says whether the service is enabled for this share.	
	Details of users and groups with readonly access.	
groups	Array of users.	
name	Name of the user	
readonly	Unique id of the user	
Array of groups.		
name	Name of the group	
id	Unique id of the group	
	Details of users and groups with full access.	
groups	Array of users.	
cifs/afp/ webdav	Name of the user	
fullaccess	Continued on next page	

Table 6.1 – continued from previous page

	Name	Value
id	Unique id of the user	
Array of groups.		
name	Name of the group	
id	Unique id of the group	
groups	Details of users and groups with no access.	
name	Array of users.	
id	Name of the user	
id	Unique id of the user	
Array of groups.		
name	Name of the group	
id	Unique id of the group	
enabled	Details of FTP services in the share.	
	Boolean value which says whether FTP service is enabled for this share.	
groups	Array of users.	
name	Name of the user	
id	Unique id of the user	
Array of groups.		
name	Name of the group	
id	Unique id of the group	
enabled	Details of NFS services in the share.	
	Boolean value which says whether NFS service is enabled for this share.	
readonly	Array of IP addresses with readonly permissions.	
readwrite	Array of IP addresses with readwrite permissions.	
recycle_bin	Details of Recycle Bin in the share.	
enabled	Boolean value which says whether service is enabled for this share.	
media_service	Details of Media Services in the share.	
enabled	Boolean value which says whether service is enabled for this share.	

6.2 GET Shares

Returns [Share object](#) containing the shares present in the NAS device.

Resource URL http://<nas_box_ip_address>/index.php/shares/api

Input → None

Response → Array of [Share objects](#)

6.3 POST - Create Share

Takes [Share object](#) containing the share to create.

Resource URL → http://<nas_box_ip_address>/index.php/shares/api

Input → [Share object](#)

Response → [Share object](#) which got created with a new *id*.

6.4 DELETE - Delete Share

Takes [Share object](#) containing the share to delete.

Resource URL -> <nas_box_ip_address>/index.php/shares/api

Input -> [Share object](#)

Response -> [Share object](#) which got deleted.

6.5 PUT - Edit Share

Takes [Share object](#) containing the share to edit.

Resource URL -> <nas_box_ip_address>/index.php/shares/api

Input -> [Share object](#)

Response -> [Share object](#) which got edited.

USERS AND GROUPS

The users and groups api exposes interfaces related to users and groups present in the the NAS device.

7.1 User Object

The following represents a user object:

```
[
  {
    "id": "",
    "name": "",
    "fullname": "",
    "email": "",
    "password": "",
    "admin": true,
    "groups": [
      {
        "id": "",
        "name": ""
      }
    ]
  }
]
```

User Object Details

Name	Value
id	The primary key unique id by which the user can be identified
name	The name of the user
fullname	Full name of the user
email	Email of the user
password	Password of the user
admin	Boolean value which says whether the user has administrative privileges.
groups	Brief info of the groups on which the user belongs to.
id	Name of the group.
id	Unique primary key id of the group.

7.2 GET Users

Returns [User object](#) containing the users present in the NAS device.

Warning: Password of the users should **NOT** be returned.

Resource URL http://<nas_box_ip_address>/index.php/users/api

Input → None

Response → Array of [User objects](#)

7.3 POST - Create User

Takes [User object](#) containing the user to create.

Resource URL → <nas_box_ip_address>/index.php/users/api

Input → [User object](#)

Response → [User object](#) which got created with a new *id*.

7.4 DELETE - Delete User

Takes [User object](#) containing the user to delete.

Resource URL → <nas_box_ip_address>/index.php/users/api

Input → [User object](#)

Response → [User object](#) which got deleted.

7.5 PUT - Edit User

Takes [User object](#) containing the user to edit.

Resource URL → <nas_box_ip_address>/index.php/users/api

Input → [User object](#)

Response → [User object](#) which got edited.

7.6 Group Object

The following represents a group object:

```
[
  {
    "id": "",
    "name": "",
    "description": "",
    "users": [
      {
        "id": "",
        "name": ""
      }
    ]
  }
]
```

```
}
]
```

Group Object Details

Name	Value
id	The primary key unique id by which the group can be identified.
name	The name of the group.
description	Description of the group.
	Brief info of the users which the group contains.
nasid	Name of the user.
id	Unique primary key id of the user.

7.7 GET Groups

Returns [Group object](#) containing the groups present in the NAS device.

Resource URL http://<nas_box_ip_address>/index.php/groups/api

Input → None

Response → Array of [Group objects](#)

7.8 POST - Create Group

Takes [Group object](#) containing the group to create.

Resource URL → <nas_box_ip_address>/index.php/groups/api

Input → [Group object](#)

Response → [Group object](#) which got created with a new *id*.

7.9 DELETE - Delete Group

Takes [Group object](#) containing the group to delete.

Resource URL → <nas_box_ip_address>/index.php/groups/api

Input → [Group object](#)

Response → [Group object](#) which got deleted.

7.10 PUT - Edit Group

Takes [Group object](#) containing the group to edit.

Resource URL → <nas_box_ip_address>/index.php/groups/api

Input → [Group object](#)

Response → [Group object](#) which got edited.

NETWORK

The network api exposes interfaces related to network present in the the NAS device.

8.1 Network Object

The following represents a network object:

```
[
  {
    "id": "eth0",
    "name": "LAN 1",
    "active" : true,
    "dhcp": true,
    "ip" : "192.168.0.10",
    "netmask" : "255.255.255.0",
    "default_gw": "192.168.0.1",
    "dns" : {
      "auto": true,
      "servers": [""]
    },
    "aggregation": {
      "enabled": true,
      "mode": ""
    },
    "mtu" : 1500,
  }
]
```

Network Object Details

Name	Value
id	The primary key unique id by which the network can be identified. Can be one of the two val
name	The name of the network.
active	Boolean value which says whether the current interface is enabled or not.
dhcp	Boolean value which says whether IP address is dhcp or static.
ip	IP address of the interface.
netmask	Netmask of the interface.
default_gw	Default gateway of the interface.
dns	DNS server details of the interface.

auto | **Boolean value which says whether dns** servers should be applied automatically.

servers **Array** containing IP addresses of DNS servers.

aggregation	Details of aggregation/bonding of the interfaces.	
-------------	---	--

enabled **Boolean** value which says whether the service is enabled for this interface.

mode Details of aggregation bonding mode. Can be one of the two values.

- roundrobin
- failover

mtu	Maximum transmission unit value.	
-----	----------------------------------	--

8.2 GET Network

Returns [Network object](#) containing the interfaces present in the NAS device.

Resource URL http://<nas_box_ip_address>/index.php/network/api

Input -> None

Response -> Array of [Network objects](#)

8.3 PUT - Edit Network

Takes [Network object](#) containing the interface to edit.

Resource URL -> <nas_box_ip_address>/index.php/network/api

Input -> [Network object](#)

Response -> [Network object](#) which got edited.

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*