

Cloudsync via rclone and a napp-it ZFS Storageserver

Setup and First steps

published: 2020-Jun-13(c) napp-it.org

Licence:
CC-BY-SA see <http://creativecommons.org/licenses/by-sa/2.0/>

Content:

1. Napp-it and S3 Cloudstorage via minIO
 - 1.1 Install minIO and rclone
2. Enable a filesystem for Cloudaccess via S3
3. Access an S3 Share via browser
4. Access an S3 share via GUI tools
5. Access cloudstorage via rclone
 - 5.1 Amazon S3 /minIO
 - 5.2 Google Gdrive
6. Security
 - 6.1 https encrypted transfer
 - 6.2 Encrypted files
7. Cloudsync as a job
8. Manuals and Questions

1. Napp-it and Cloudstorage

Local Storage

You mainly use napp-it as a ZFS fileserver on your LAN to share ZFS filesystems via FC/iSCSI, NFS and SMB. You can work directly on the filer with multiuser filelocking and permission restrictions and a performance that is similar to your local disk.

Cloud Storage

You can use napp-it to share ZFS filesystems to the Internet with the Amazon S3 protocol and minIO. This is an Amazon S3 kompatible OpenSource server application that is supported by napp-it and OmniOS extra repo, <https://min.io/product/overview>. An S3 share is accessible from the Internet via browser or one of the Amazon S3 compatible backup and sync tools (Google: s3 sync tool). When you enable an S3 share, you must enter a location, a login with password and the port over which you want to access the service. You can use http or preferable https.

To install minIO on OmniOS: goto menu Services > S3

Cloud Backup and Cloud Sync

If you want to access files on the Internet or from a Cloud service provider you cannot work directly on the cloudstorage but must work locally as the Internet performance is much lower than a LAN. You use the Cloud for backup or to sync and share local files via a service provoder like Amazon S3, Dropbox, Google, Microsoft or can use your local ZFS system as an inhouse cloudstorage. To sync files from your local system (Desktop, Laptop or a napp-it storage server) you can use sync or backup tools. If you want to copy or sync files from your napp-it server to a cloud storage, you can use rclone, <https://rclone.org/>

You can use rclone to sync or backup files from/to a local system and from/to a cloudstorage or between cloudservices.

1.1 Install minIO and rclone manually

MinIO and rclone are provided by the OmniOS extra repository.

Install via pkg (or napp-it menu Services > S3)

```
pkg set-publisher-g https://pkg.omniosce.org/r151032/extra extra.omnios
```

```
pkg install minio
pkg install minio-mc
pkg install rclone
```

```
#optionally enable minio as a service
# (I prefer a manual start with options, you can start multiple instances with a different port)
svcadm enable minio
```

```
# service manifest, see /lib/svc/manifest/application/application-minio.xml
```

2. Enable a ZFS filesystem for S3 internet access

Open menu ZFS filesystems and click under S3_cloud (unset) to enable a S3 share

hfg/test: shareS3 X

Ändern der Eigenschaft hfg/test/: shareS3

REGION_NAME (opt)	my_location	<input type="button" value="set property"/>
ACCESS_KEY (>2)		
ACCESS_SECRET (>7)		
ACCESS_KEY_OLD		
ACCESS_SECRET_OLD		
Cert folder	/var/web-gui/_log/minio	
ADDRESS:PORT	:9014	
S3 key backupfolder	hfg/test v	
Clear/recreate minio settings	no v	
Minio server	/hfg/test/S3_data	
Alt minio server		
Nice	yes v	
Autostart	no v	

S3 cloudservices are provided via minIO

*You need ACCESS_KEY and ACCESS_SECRET (name,password) to access the S3 folder from a client via an Internet browser or another S3 client on the server ip:port ex 192.168.1.1:9000
To change a key and secret later, disable the S3 share and re-enable with a new KEY and SECRET.*

*Per default you share S3 data in a subfolder S3_data with settings in a subfolder S3_config.
You can override the default minio server call with the alt minio server field.*

*Per default S3 access is done on any ip via port 9000. You can set the port via :9001 or 192.168.1.1:9003
If you share several ZFS filesystems, each minIO session needs an unique ip:port.*

*MINIO_REGION_NAME (string) name of the location of the server e.g. "us-west-rack2"
autostart: on/of after reboot*

alt server: set your own server options for minio The cert functionality is is enabled when you provide /var/web-gui/_log/minio/public.crt.

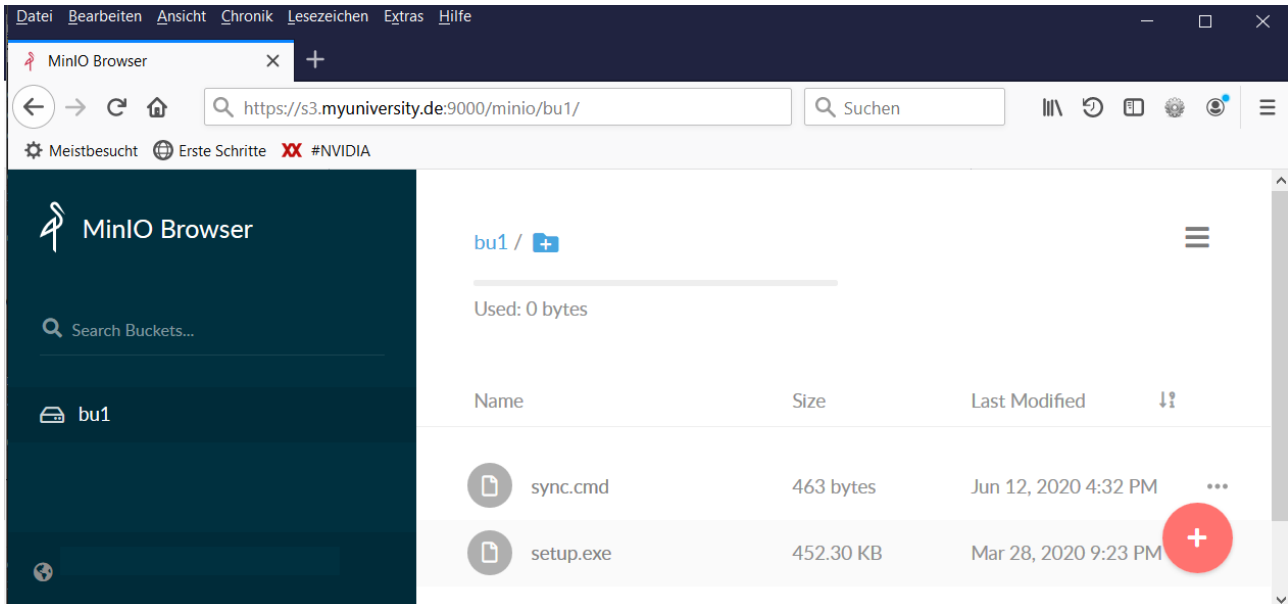
*With nice, you can reduce priority of minio compared to other storage actions.
Calculate at least additional 400 MB RAM per minIO session. Data of an S3 are stored in a subfolder S3_data.
or at a default location that you can define in About > settings. Optionally clear and recreate settings and key.*

Enter REGION_NAME, ACCESS_KEY, ACCESS_SECRET and port.
You need them later for access.

3. Access an S3 share via browser

Enter the hostname and the port into your browser.

MiniIO will ask for access key and secret password. You can then access the ZFS filesystem from elsewhere and upload/download files or create an anonymous link for a file (valid up to seven days)



4. Access an S3 share via GUI tools

To access an S3 share, you can use GUI backup tools ex Duplicati or GUI sync tools like CloudBerry, CyberDuck, S3 Browser, DragonDisk or Arq.

5. Access an S3 share (or other cloudservices) via rclone (CLI tool)

Rclone is in the OmniOS extra repo and installed with minIO in menu Services > S3. You find the binaries in `/opt/ooce/`. To pass parameters to rclone, you can create a script where you pass parameters via export ex (use the parameter from your napp-it S3 share) You can use any parameter from a rclone config file, just use it with uppercase char ex region -> REGION

```
#!/usr/bin/sh
export RCLONE_CONFIG_MYS3_TYPE=s3
export RCLONE_CONFIG_MYS3_ACCESS_KEY_ID=loginname
export RCLONE_CONFIG_MYS3_SECRET_ACCESS_KEY=secetpassword
export RCLONE_CONFIG_MYS3_ENDPOINT=https://s3.university.org:9000
export RCLONE_CONFIG_MYS3_REGION=my_location
```

```
#sync (a bucket (folder) bu1 must exist on your S3 share for this example
/opt/ooce/bin/rclone copy /anyfolder/ MYS3:/bu1
```

```
#list files in folder bu1
/opt/ooce/bin/rclone ls MYS3:/bu1
```

5. Config rclone for any Cloudservice

Start interactive setup for rclone, https://rclone.org/commands/rclone_config/
`# /opt/ooce/bin/rclone config` and select and configure a Cloud service

Tips

If you want to configure a remote where you want to copy/paste from console: use Putty (as root) and copy/pasted from/to console with a mouse right click.

5.1 Config rclone for Amazon S3 compatible (minIO)

Create one or more remote shares with options. To access a napp-it S3 share, you need at least the parameters `type`, `provider`, `access_key`, `secret_key`, `endpoint` and `region`

When rclone is configured, you can get the location of the config file via `/opt/ooce/bin/rclone config` file

example: config file: `/root/.config/rclone/rclone.conf`

Check config file via
`cat /root/.config/rclone/rclone.conf`

```
[cloud]
type = s3
provider = Minio
#env_auth = false
access_key_id = loginname
secret_access_key = secretpassword
endpoint = https://s3.university.org:9000
region = my_location
```

You can now use `cloud:` as source or destination in rclone commands like `copy` or `sync`

ex enter at console:
`/opt/ooce/bin/rclone ls cloud:`

If you want to start rclone as a job, enter the rclone command as jobaction or call a script. You can enter several ongoing commands `cmd1; cmd2; etc` for a single line multi-command. If you want to keep a local filesystem in sync with a remote filesystem, use `sync`. This will delete remote files no longer in the local folder.

If you want to have newer files synced both ways, start a local copy -> Cloud followed by a copy Cloud > local with `-u` (newer files only). To delete files then you must delete locally and remote. You can not only sync napp-it ex to/from S3 or Google, you can also sync Google with Amazon S3 or compatible.

5.2 Config rclone for Google Gdrive

<https://github.com/Cloudbox/Cloudbox/wiki/Google-Drive-API-Client-ID-and-Client-Secret>

Login into Google (Chrome), Connect to <https://console.developers.google.com/apis/>

- create a project rclone
- enable Google Drive API
- create credentials OAuth with additional drive.file API (OAuth Client ID and other or TV and limited Input devices)
This give the client id and client secret

Start rclone config

- create a new remote (Google drive) ex gdrive:
- enter client id and secret from above
- For verification, select headless mode
this displays a verification link
- Copy/paste the verification link into a browser and allow access
This will return a verification value
- Enter the verification value into rclone config and save

Now verify:

```
rclone ls gdrive
```

This should gave a listing of your Google drive content

6. Security

6.1 Encrypted transfer with https

To enable https for minIO, place a public and private key in `/var/web-gui/_log/minio/`

- private.key
- public.crt

Content of private.key

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAt//67pLLarmyUldJet+YzISJhI9w9DD8OK2A61NODIVztv3V
.....
iy6a+6g7uzR6/RISPWvvC05SXZxeMu/uMx7svtC9z88QfmVYWNDI5Q==
-----END RSA PRIVATE KEY-----
```

Content of public.crt

```
subject=/CN=s3.university.de/OU=it/O=University...
-----BEGIN CERTIFICATE-----
MIIEpAIBA....
```

6.2 Encrypt/ Decrypt files

Start minio config and create two entries:

s3cloud (the regular unencrypted cloud) and an encrypted device that points to the unencrypted remote and enc_s3: the device that does the encryptions and forward to s3cloud:

examples:

- name: s3cloud
- type: s3
- provider: Minio
- key: xxxx
- secret: yyyyyyyyyy
- endpoint: http(s)://xxx.yy:9000
- region: my_location (from s3 share settings)

and

- name: enc_s3 (use a name that identify the crypt property and the undelying remote)
- type of storage: 10 (enc/decrypt a remote)
- remote: s3:/bucket1 (s3: and s3:/bucket1 must exist)
- pw: 22222222 (enter twice)
- enc file and directoty names: optional
- salt (optional)

Save the password and the rclone.conf file with the hashed password and settings at a save place !!!

result: (rclone.conf) ex:

```
[s3cloud]
type = s3
provider = Minio
access_key_id = loginname
secret_access_key = wnkiPWs3#+
endpoint = https://s3.hfg-gmuend.de:9000
region = my_location

[enc_s3]
type = crypt
remote = s3cloud:/bu2
filename_encryption = standard
directory_name_encryption = true
password = HW8linj-vFk4XFXNF0v-ii2p100wSgr3
```

The above password is a hash value. Save the config file with your settings at a save place

You can now upload files to enc_s3:

This will save the files encrypted according the settings (a bucket ex bu2 must exist)
`/opt/ooce/bin/rclone copy /folder/ enc_s3:`

or download and decrypt:

`/opt/ooce/bin/rclone copy /folder/ enc_s3: /folder/`

**Save your passwords and rclone config file on a save place. On Problems:
 Restore the config file (or the needed remote entry) from backup**

<code>rclone config file</code>	returns path to config file
<code>rclone config show</code>	returns content of config file

7. Cloudsync as a job (napp-it „other job“)

Test a command at CLI or create a script with the wanted actions ex sync.sh:

```
#!/bin/sh
/opt/ooce/bin/rclone copy -u /pool/filesystem gdrive:/backup1
```

If you want to have new files locally and cloud (delete must be done manually both sides)
`/opt/ooce/bin/rclone copy -u /pool/filesystem gdrive:/backup1`
`/opt/ooce/bin/rclone copy -u gdrive:/backup1 /pool/filesystem`

If the action works as expected, create an other job with the command ex
`/opt/ooce/bin/rclone copy -u /pool/filesystem gdrive:/backup1`
 or a script ex „sh /path/script.sh“ as action.

8. Manuals and Questions

https://napp-it.org/manuals/index_en.html

<https://forums.servethehome.com/index.php?threads/amazon-s3-compatible-zfs-cloud-with-minio.27524/>