LibreS3 Quick Start Guide

1

INTRODUCTION

LibreS3 is a robust Open Source implementation of the Amazon S3 service, supporting a subset of the S3 REST API.

Standard S3 client libraries and tools can be used to access it (for example s3cmd, python-boto, DragonDisk, etc.) .

LibreS3 uses Skylable S^X as the storage backend, which provides data deduplication and replication.

USEFUL LINKS

- http://lists.skylable.com

- https://bugzilla.skylable.com

- http://wiki.skylable.com

- http://www.skylable.com/products/libres3

INSTALLATION

LibreS3 is regularly tested on Linux and FreeBSD. We recommend using the binary
packages from http://www.skylable.com/download if your platform is supported.

BINARY PACKAGES

DEBIAN WHEEZY

Add the following entry to /etc/apt/sources.list.d/skylable.list:

```
deb http://cdn.skylable.com/debian wheezy main
```

then run the following commands:

```
# wget https://pgp.mit.edu/pks/lookup?op=get&search=0x5377E192B7BC1D2E |
    sudo apt-key add -
# apt-get install libres3
```

CENTOS 6/7

Create the file /etc/yum.repos.d/skylable-sx.repo with this content:

```
[skylable-sx]
name=Skylable SX
baseurl=http://cdn.skylable.com/centos/$releasever/$basearch
enabled=1
gpgcheck=0
```

then execute:

```
# yum install libres3
```

FEDORA 20

Create the file /etc/yum.repos.d/skylable-sx.repo with this content:

```
[skylable-sx]
name=Skylable SX
baseurl=http://cdn.skylable.com/fedora/$releasever/$basearch
enabled=1
gpgcheck=0
```

then execute:

```
# yum install libres3
```

SOURCE CODE

On most Unix platforms you can compile LibreS3 from source. You will need the following packages to be installed together with their development versions:

- OCaml (>= 3.12.1)

- camlp4 (matching your OCaml compiler version)

- OpenSSL

- PCRE C library

- GNU Make and m4

For example, on Debian run:

```
# apt-get install ocaml-native-compilers camlp4-extra libssl-dev libpcre3-
    dev make m4
```

On Fedora run:

```
# yum install ocaml /usr/bin/camlp4of /usr/bin/camlp4rf /usr/bin/camlp4 pcre
    -devel openssl-devel make m4 ncurses-devel
```

COMPILATION

Follow the standard installation procedure. To install LibreS3 into the default location (`/usr/local`):

```
$ ./configure && make && make check
# make install
```

The rest of the manual assumes that LibreS3 was installed from a binary package, so some paths may be different.

Note: On OpenSolaris/OmniOS you need some additional flags:

```
$ CPPFLAGS=-m64 ./configure --destdir=${DESTDIR} && make && make check
# make install
```

CONFIGURATION

REQUIREMENTS

LibreS3 by default listens on ports 8008 and 8443, which need to be available on a given IP address.

LibreS3 connects to the SX cluster via HTTP(S). You can run LibreS3 and SX on the same or different hosts.

DNS ZONE ENTRY

S3 buckets require a wildcard A record pointing to the IP address (1.2.3.4 below) of the host running LibreS3, for example:

```
*.libres3.example.com. A 1.2.3.4
libres3.example.com. A 1.2.3.4
```

In case you don't have control over the DNS you'll have to modify the /etc/hosts file of each client machine and add a line for each bucket you want to access:

```
libres3.example.com 1.2.3.4
bucket1.libres3.example.com 1.2.3.4
bucket2.libres3.example.com 1.2.3.4
```

SETTING UP A LIBRES3 NODE

Setting up LibreS3 is as simple as running the interactive tool libres3_setup. If you provide the path to an existing S^X cluster configuration file created by sxsetup, most of the settings will be done automatically.

Please make sure the default volume replica count setting is less or equal to the number of nodes in the SX cluster.

In the examples below we assume you have an S^X cluster running and you want to run LibreS3 on libres3.example.com.

Example setup with sxsetup.conf:

```
# libres3_setup --sxsetup-conf /etc/sxserver/sxsetup.conf
Successfully loaded SX configuration from '/etc/sxserver/sxsetup.conf'

S3 (DNS) name: libres3.example.com

Generating default SSL certificate and key in /etc/ssl/certs/libres3.pem and
    /etc/ssl/private/libres3.key
Generating a 2048 bit RSA private key
.......................................+++
.............+++
```

```
writing new private key to '/etc/ssl/private/libres3.key'
-----

S3 HTTPS port: 8443

S3 HTTP port: 8008

Default volume size [use K, M, G and T suffixes]: 100G

Default volume replica count: 1

Generating '/etc/libres3/libres3.conf'
Updating '/etc/libres3/libres3.sample.s3cfg'
Generating '/etc/libres3/libres3.sample.boto'
Updating '/etc/libres3/libres3-insecure.sample.s3cfg'
Generating '/etc/libres3/libres3-insecure.sample.boto'

Do you want to start LibreS3 now? [Y/n] Y
[....] Restarting LibreS3: libres3No libres3_ocsigen found running; none
    killed.

Loading configuration from /etc/libres3/libres3.conf
Waiting for server to start (5s) ... OK
. ok
```

Example without sxsetup.conf: [1]

```
# libres3_setup
Admin key or path to key-file: ODPiKuNIrrVmD8IUCuw1hQxNqZfJOhlBUgyckAolodd4C
    /4r4ecY3QAA

SX server IP/DNS name: sx.example.com

SX server HTTPS port: 443

Run as user: nobody

Run as group: nogroup

S3 (DNS) name: libres3.example.com

Generating default SSL certificate and key in /etc/ssl/certs/libres3.pem and
    /etc/ssl/private/libres3.key
Generating a 2048 bit RSA private key
.......................................................+++
............+++
writing new private key to '/etc/ssl/private/libres3.key'
-----

S3 HTTPS port: 8443

S3 HTTP port: 8008

Default volume size [use K, M, G and T suffixes]: 100G

Default volume replica count: 1

Generating '/etc/libres3/libres3.conf'
File '/etc/libres3/libres3.conf' already exists, overwriting
Updating '/etc/libres3/libres3.sample.s3cfg'
Generating '/etc/libres3/libres3.sample.boto'
Updating '/etc/libres3/libres3-insecure.sample.s3cfg'
Generating '/etc/libres3/libres3-insecure.sample.boto'

Do you want to start LibreS3 now? [Y/n] Y
[....] Restarting LibreS3: libres3No libres3_ocsigen found running; none
    killed.

Loading configuration from /etc/libres3/libres3.conf
Waiting for server to start (5s) ... OK
```

---

[1] you can use 'sxadm node --info /var/lib/sxserver/storage/' on the SX node to find out the required information

```
.  ok
```

To start/stop LibreS3[2]:

```
# libres3 start
Starting LibreS3
LibreS3 started successfully
# libres3 status
--- LibreS3 STATUS ---
LibreS3 is running (PID 28245)

--- LibreS3 INFO ---
SSL private key: /etc/ssl/private/libres3.key
LibreS3 logs: /var/log/libres3/
# libres3 stop
Loading configuration from /etc/libres3/libres3.conf
Sending TERM to PID 28245 ...
Waiting for PID 28245 ...
```

If the server doesn't start, please check the log files for details.

That's it - your LibreS3 cloud storage is already up and running! You can now connect to it with your favorite S3 client.

---

[2]LibreS3 and SX will communicate using SSL by default. For debugging purposes you can configure SX with 'sxsetup –no-ssl' and then you have to start LibreS3 with –no-ssl

CLIENT CONFIGURATION

S3CMD

You can use the generated s3cfg config file [1] or configure s3cmd from scratch. Below we assume that your LibreS3 is running on `"libres3.example.com"` and it supports SSL. The important s3cmd configuration settings are:

```
use_https True
host_base libres3.example.com:8443
host_bucket %(bucket)s.libres3.example.com:8443
access_key <your-sx-username>
secret_key <your-sx-key>
ca_certs_file s/etc/ssl/certs/libres3.pem
```

In case you don't use SSL, please use the port 8008 instead of 8443, and set `"use_https"` to `"False"`. Once you've configured s3cmd check that it properly connects to LibreS3:

```
$ s3cmd ls --debug 2>&1 | grep host
```

Supported s3cmd commands:

Bucket `mb`, `rb`, `ls`, `la`, `du`, `info`, `setpolicy`, `delpolicy`, `multipart`

Object `put`, `get`, `del`, `sync`, `info`, `cp`, `modify`, `mv`, `abortmp`, `listmp`, `signurl`

ENCRYPTED FILES

You can set `gpg_passphrase` in `.s3cfg` and use `s3cmd --encrypt` to upload/-download encrypted files.

CAVEATS

Certificate verification changed in Python version 2.7.9+, and you'll need an s3cmd version newer than 1.5.0-rc1 to match. The .s3cfg must contain a `ca_certs_file` entry pointing to the certificate of the LibreS3 server, otherwise certificate verification (and thus HTTPS connections will fail). Note that wildcard SSL certificates only match one level, i.e. a certificate for `*.s3.example.com` is valid for `a.s3.example.com`, but it is NOT valid for `a.b.s3.example.com`, hence you avoid using bucket names which contain dots.

---

[1] `/etc/libres3/libres3.sample.s3cfg`

PYTHON-BOTO

S3 clients using Python boto are configured in `~/.boto`, or you can use the generated [2] file. A typical configuration looks as follows:

```
[Credentials]
aws_access_key_id=<your-sx-username>
aws_secret_access_key=<your-sx-key>
s3_host=libres3.example.com
s3_port=8443
[Boto]
is_secure = True
```

Note that setting `"s3_host"` will override the hostname you give to applications on the command-line. If you are using an application that allows setting the S3 hostname on the command-line, you might want to use that instead. Note that old version of python-boto require the port to be on `s3_host` instead of `s3_port`.

S3FS-FUSE

S3FS can be used to provide a FUSE-based file system backed by LibreS3.

You must specify the SX username and access token in `~/.passwd-s3fs`, the URL for your LibreS3 server with `-o url`, and the certificate used by LibreS3 in `CURL_CA_BUNDLE`:

```
$ cat >~/.passwd-s3fs <<EOF
admin:0DPiKuNIrrVmD8IUCuw1hQxNqZfJOhlBUgyckAolodd4C/4r4ecY3QAA
EOF
$ chmod 0600 ~/.passwd-s3fs
$ mkdir ~/libres3-vol1
$ CURL_CA_BUNDLE=/etc/ssl/certs/libres3.pem s3fs -o url=https://libres3.
    example.com:8443 vol1 ~/libres3-vol1 -o uid=1000
```

If you want to access LibreS3 unencrypted for debugging purposes then:

```
$ s3fs -o url=http://libres3.example.com:8008 vol1 ~/libres3-vol1 -o uid
    =1000
```

CAVEATS

If s3fs fails to connect to LibreS3 it quits without an error message, and the next time you access the mountpoint you get an error, including trying to start s3fs again:

```
$ touch ~/libres3-vol1/x
touch: cannot touch '/home/USER/libres3-vol1/x': Transport endpoint is not
    connected
$ s3fs ...
s3fs: unable to access MOUNTPOINT /home/USER/libres3-vol1: Transport
    endpoint is not connected
$ fusermount -u ~/libres3-vol1
$ s3fs ...
```

You'll have to manually unmount and it is useful to run `s3fs` in the foreground to see the actual error message:

```
$ fusermount -u ~/libres3-vol1
$ CURL_CA_BUNDLE=/etc/ssl/certs/libres3.pem s3fs -o url=https://libres3.
    example.com:8443 vol1 ~/libres3-vol1 -o uid=1000 -f
[...]
```

---

[2]`/etc/libres3/libres3.sample.boto`

Common errors are using the wrong user/token in /.passwd-s3fs, or s3fs reading the wrong passwd-s3fs (from /etc for example), or that you don't have permissions to access the volume, etc.

OTHER CLIENTS

For information on other clients please refer to our wiki: `http://wiki.skylable.com/wiki/LibreS3_Clients`

PUBLIC ACCESS TO OBJECTS

SIGNED URLS

To create a signed URL valid for 24 hours for retrieving the object x from the bucket vol1:

```
# s3cmd -c /etc/libres3/libres3.sample.s3cfg signurl s3://vol1/x `date -d
    '24 hours' +%s` | sed -e 's/http:/https:/'
https://vol1.libres3.example.com:8443/x?AWSAccessKeyId=admin&Expires
    =1421780366&Signature=uT1uzWiRKOcr%2F459zjLvmWoMTSg%3D
```

The URL can be pasted into a browser, and used with curl or wget to download the file without additional authorization. After 24 hours the URL expires and can't be used anymore.

There is also a python script as a sample for writing your applications that would generate signed URLs:

```
$ git clone http://git.skylable.com/experimental
$ cd experimental/s3genlink
# BOTO_CONFIG=/etc/libres3/libres3.sample.boto ./s3genlink.py vol1/x
```

PUBLIC READ-ONLY ACCESS TO OBJECTS

Signing URLs can be inconvenient if you want to make all objects in a bucket publicly accessible. In this case you can enable public access to all objects in a bucket by setting a bucket policy (replace vol1 with the name of your bucket):

```
$ cat >anon.json <<EOF
{"Version":"2012-10-17",
 "Statement" : [{ "Sid":"AddPerm",
    "Effect": "Allow", "Principal": "*",
    "Action": ["s3:GetObject"],
    "Resource":["arn:aws:s3:::vol1/*"]}]}
EOF
# s3cmd -c /etc/libres3/libres3.sample.s3cfg setpolicy anon.json s3://vol1
# s3cmd -c /etc/libres3/libres3.sample.s3cfg info s3://vol1
[...]
$ wget vol1.libres3.example.com:8008/x
$ wget libres3.example.com:8008/vol1/x
```

To disable public access just delete the policy:

```
# s3cmd -c /etc/libres3/libres3.sample.s3cfg delpolicy s3://vol1
```

Note that for public/anonymous users only downloading individual objects is permitted – listing the bucket's contents is not.

There are some scripts to help managing public buckets:

```
$ git clone http://git.skylable.com/experimental
$ cd experimental/s3publicvol
# BOTO_CONFIG=/etc/libres3/libres3.sample.boto ./s3publicvol.py vol1/x
Using access key id: admin
Turning on public access for bucket vol1
# BOTO_CONFIG=/etc/libres3/libres3.sample.boto ./s3policy.py vol1/x
Using access key id: admin
Bucket vol1 has an access policy:
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::vol1/*"
    }
  ]
}
# BOTO_CONFIG=/etc/libres3/libres3.sample.boto ./s3privatevol.py vol1/x
Using access key id: admin
Turning off public access for bucket vol1
# BOTO_CONFIG=/etc/libres3/libres3.sample.boto ./s3policy.py vol1/x
Using access key id: admin
Bucket vol1 is private
```

Note that using ACLs to set individual objects as publicly accessible is not supported.

SETTING CONTENT-TYPE

An object's Content-Type on download will match the value used on upload via LibreS3. If you use s3cmd to upload or copy your files the Content-Type will usually be guessed correctly already, but you can override it with `--mime-type=`. For files uploaded via SX the Content-Type is guessed automatically based on file extension.

5

TROUBLESHOOTING

If you face a problem connecting to your LibreS3 server, please check the log files located at:

`/var/log/libres3/*.log`

You can also enable logging of full HTTP requests with:

```
# export LIBRES3_DEBUG=1
# libres3 restart
```

For more information and FAQ please visit http://wiki.skylable.com.

If you can't find your solution there, please subscribe to our mailing list at http://lists.skylable.com and post about your issues.