

Installation Guide

[Source code \(main branch\) and documentation](#) is available on Github under Apache Licenses, Version 2.0.

This page contains some additional guides and hints provided by users installing the platform:

- [Staszek Jankowski, Maciej Brzeniak \(PSNC\)](#) - Centos 5.8
- [Dirk Dupont \(BELNET\)](#) - Ubuntu 10.04 32 bit
- [Jakub Peisar \(CESNET\)](#) - Ubuntu 12.04 64 bit single and multi VMs
- [Guilherme Maluf Balzana \(RNP\)](#) - TCD with Open Swift storage

PSNC - TRUSTED CLOUD DRIVE INSTALLATION INSTRUCTION

Tested configuration:

Centos 5.8
Oracle Java 1.7.0_07-b10

This is a single node installation. All components are installed on the same node.

Instruction:

1. Unzip archive:

```
unzip VirtualCloudDrive-CloudDrive-88d54c2.zip
```

2. Run voldemort:

```
cd VirtualCloudDrive-CloudDrive-88d54c2/binaries/metadata/voldemort-0.90.1-patched/bin  
./voldemort-server.sh ../config/clouddrive_node_cluster/ &
```

3. Install and setup MySQL:

```
yum install mysql-server mysql  
/etc/init.d/mysqld start
```

Set root password:

```
mysqladmin -u root password "newpassword"  
mysqladmin -u root -h host_name password "newpassword"  
/etc/init.d/mysqld restart
```

Create database:

```
mysql -u root -p  
create database rightfabric;  
quit
```

4. Setup and run virtual clouddrive webdav module:

```
mkdir /etc/rightfabric  
cd ../VirtualCloudDrive-CloudDrive-88d54c2/config/clouddrive  
cp config.txt /etc/rightfabric/config.txt
```

```
vi /etc/rightfabric/config.txt
```

```
change: voldemort = tcp://10.0.0.5:6666  
to: voldemort = tcp://localhost:6666
```

```
cd ../VirtualCloudDrive-CloudDrive-88d54c2/src/clouddrive  
./sbt console  
:load ./addme.scala  
:quit  
./sbt run
```

test with cadaver:

```
cadaver localhost:9090  
user: maarten  
pass: geheim
```

5. Setup clouddrive website:

Replace the value of **db.password** with mysql password on *src/main/resources/props/default.props* file

```
cd ../VirtualCloudDrive-CloudDrive-88d54c2/src/web_clouddrive  
./sbt update  
./sbt ~jetty-run
```

Open <http://localhost:8080> in your favourite webbrowser and enjoy!

Authors:

- Staszek Jankowski (staszek ---at--- man.poznan.pl)
- Maciej Brzeniak (maciekb ---at--- man.poznan.pl)

TODO:

- the instruction to be cross-checked by someone else than the author
- to be checked for other popular distributions

BELNET - TRUSTED CLOUD DRIVE INSTALLATION INSTRUCTION**Tested configuration:**

Ubuntu 10.04 32bit

Single node setup

Pre-installations:

```
apt-get install mysql-server
```

root password must be geheim

(geheim is Dutch for secret)

```
apt-get install openjdk-6-jdk
```

Credits:

Dirk Dupont (dirk.dupont-at-belnet.be)

CESNET - TRUSTED CLOUD DRIVE INSTALLATION INSTRUCTIONS**A) Single VM installation:**

All virtual machines were Ubuntu 64-bit 12.04 LTS with OpenJDK 1.7

For a single VM installation I followed the [PSNC guide](#)

I didn't bother with setting up SimpleSamI PHP and Apache, I was using just Jetty to serve the webcontent.

```
cloud@clouddriveApp1:~$ uname -a
```

```
Linux clouddriveApp1 3.2.0-32-generic #51-Ubuntu SMP Wed Sep 26 21:33:09 UTC 2012 x86_64 x86_64 x86_64 GNU/Linux
```

```
cloud@clouddriveApp1:~/CloudDrive-master$ java -version
```

```
java version "1.7.0_09"
```

```
OpenJDK Runtime Environment (IcedTea7 2.3.3) (7u9-2.3.3-0ubuntu1~12.04.1)
```

```
OpenJDK 64-Bit Server VM (build 23.2-b09, mixed mode)
```

Only issue I found was with the jquery.js paths (see below)

B) Multiple VMs installation:

I didn't bother with setting up SimpleSamI PHP and Apache, I was using just Jetty to serve the webcontent.

For the multiple VMs installation I followed the <https://github.com/VirtualCloudDrive/CloudDrive/wiki/Two-Quick-starts>

I made 5 VMs for this setup :

- Voldemort1
- Voldemort2 (I wanted to try the multi-node cluster)
- App1 (WebDAV daemon and the Scala application)
- MySQL (distinct VM for the database)
- Webserver1 (just a one webserver VM to server the traffic, can easily scale)

On all machines I downloaded the source tar from github. All the machines also have the same /etc/rightfabric/config.txt file, which I edited to suit my architecture (file attached). Then on Voldemort machines I changed the CloudDrive-master/binaries/metadata/voldemort-0.90.1-patched/config/clouddrive_node_cluster/config/cluster.xml to have 2 nodes (file attached).

On the MySQL machine I obviously installed the mysql-server and set up the database for the application. For this I used the SQL script at CloudDrive-master/config/database/mysql/mysql-setup.sql.

On the App server I created the directory structure to hold the user files and compiled the WebDAV daemon with sbt. After running the daemon with "./sbt run" the daemon was up and serving requests.

This is how the "storage" looks :

```
cloud@clouddriveApp1:~$ ls -lh /cloud/data/maarten/
total 8.3M
-rw-r--r-- 1 cloud cloud 128 Nov 26 14:34 01a6cad3-a67a-4129-8457-4aafb5dceb6
-rw-r--r-- 1 cloud cloud 32 Nov 26 16:54 1b7892b2-281b-4b13-ae7e-9901a1efbba0
-rw-r--r-- 1 cloud cloud 784 Nov 26 14:51 25c61c78-4b0a-4926-8605-978b22157672
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:08 3c43d014-af5c-4a53-95e9-ae1ba6858b0c
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:08 3e0719ab-9458-41f7-a259-65882e08bebf
-rw-r--r-- 1 cloud cloud 16 Nov 28 09:35 5d1520fc-8e78-446a-a7aa-f64ac986f3a3
-rw-r--r-- 1 cloud cloud 8.3M Nov 28 11:30 927058df-09ee-44c6-a05d-8268abaf02b6
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:42 a54b85d6-2f34-4b5d-bdd8-e335365c45ea
-rw-r--r-- 1 cloud cloud 4.1K Nov 28 09:36 d481e8f2-da9f-4957-8e35-17d73eedc6f
-rw-r--r-- 1 cloud cloud 784 Nov 26 14:47 d9b19d10-1608-45fc-bdb4-5cf841660d75
```

On the webserver I decided to simply use provided Jetty to start things up. There I only edited the CloudDrive-master/src/web_clouddrive/src/main/props/production.default.props file to connect to the MySQL server on the other VM, packaged it with "./sbt package" and copied the resulting .war file to CloudDrive-master/binaries/website/root.war.

Issuing ./sbt update and then ./sbt -jetty-run I was able to start the website. You can reach it at <http://clouddrivewebsserver1.du1.cesnet.cz:8080>

Problems/Solutions:

1)

Both installations have the same problem with serving the jquery.js file. When I am logged in the website and I am at the root of my home directory view, I can click the buttons to create new folder or upload a file with no problems (buttons have javascript onclick listeners on them). The URL of jquery.js file is <http://clouddrivewebsserver1.du1.cesnet.cz:8080/jquery.js>. The URL of the webpage is <http://clouddrivewebsserver1.du1.cesnet.cz:8080/webdrive>.

But when I descend into a subdirectory, say /test/ and I try to create new folder, nothing happens. Back when I check the Jetty log I can see the 404 error and a Java stack trace of NullPointerException (can be seen in included file). My browser is now trying to access the URL <http://clouddrivewebsserver1.du1.cesnet.cz:8080/webdrive/jquery.js> while the URL of the webpage is <http://clouddrivewebsserver1.du1.cesnet.cz:8080/webdrive/test>.

Solution:

When I was using the website and I was at the root view of my directories, my browser (Chromium 20) tried to access the jquery library at example.org/jquery.js

When I descended into a directory, it tried to access jquery at example.org/webdrive/jquery.js.

And when I went into a directory inside the previous directory it tried to access it at example.org/webdrive/test/jquery.js

The directories are structured as /test/nexttest/.

It seemed the URL of jquery.js was relative to the webpage I was accessing. When I looked into the HTML code that was the case:

```
<script type="text/javascript" src="jquery.js" id="jquery"></script>
```

When I edited the template at CloudDrive-master/src/web_clouddrive/src/main/webapp/templates-hidden/ and changed it to "/jquery.js" and then redeployed the webapp it started working.

2)

The second problem is with multi-VM setup and downloading files. When I try to download a file via the website in the single-VM setup, it works and I am able to successfully download the file. But when I try to download a file via the website from the multi-VM setup, I get an error webpage and then when I check the folder, the file I tried to download is gone. Debug logging from the Jetty of the failed download attempt can be seen in the attached file.

After going through the configuration files I suspect the error being in declaration of filesystem_prefix in /etc/rightfabric/config.txt. There it states:

```
#If you use the local filesystem for data storage, this is the folder or
#mount point to store. Note that it must be shared across all machines in a
#multi-machine setup. E.g. via SMB or NFS or so.
#NOTE THE TRAILING SLASH IN THE PATH
filesystem_prefix = /cloud/data/
```

The thing is, I have this setup as a directory on the local disk of my App VM as that is the place where I store the blobs (I upload files via cadaver which connects directly to App VM). I can see the files being there:

```
cloud@clouddriveApp1:~$ ls -lh /cloud/data/maarten/
total 17M
-rw-r--r-- 1 cloud cloud 128 Nov 26 14:34 01a6cad3-a67a-4129-8457-4aafbd5dceb6
-rw-r--r-- 1 cloud cloud 32 Nov 26 16:54 1b7892b2-281b-4b13-ae7e-9901a1efbba0
-rw-r--r-- 1 cloud cloud 784 Nov 26 14:51 25c61c78-4b0a-4926-8605-978b22157672
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:08 3c43d014-af5c-4a53-95e9-aelba6858b0c
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:08 3e0719ab-9458-41f7-a259-65882e08bebf
-rw-r--r-- 1 cloud cloud 16 Nov 28 09:35 5d1520fc-8e78-446a-a7aa-f64ac986f3a3
-rw-r--r-- 1 cloud cloud 8.3M Nov 28 11:30 927058df-09ee-44c6-a05d-8268abaf02b6
-rw-r--r-- 1 cloud cloud 784 Nov 26 16:42 a54b85d6-2f34-4b5d-bdd8-e335365c45ea
-rw-r--r-- 1 cloud cloud 8.3M Nov 28 12:21 accfe894-5f04-40dc-84b9-7442bb14b27d
-rw-r--r-- 1 cloud cloud 4.1K Nov 28 09:36 d481e8f2-da9f-4957-8e35-17d73eedc6f
-rw-r--r-- 1 cloud cloud 784 Nov 26 14:47 d9b19d10-1608-45fc-bdb4-5cf841660d75
```

Then on the webserver I can see the directory is created but obviously it is empty:

```
cloud@clouddriveWebserver1:~/CloudDrive-master$ ls -lh /cloud/data/maarten/
total 0
```

Do I understand correctly, that I need to have the filesystem with binary blobs of stored files mounted even on the webserver nodes ? I would imagine the WebDAV daemon nodes having the need to see the filesystem as they process the files.

Solution:

The reason you're not seeing the data on you web server is that the actual data needs to be on a shared filesystem between the instances.

The easiest way to think of it is by keeping in mind a few things:

- 1) the metadata is in Voldemort and gives a (hopefully) consistent view of everything without visiting the actual file-store. This speeds up and provides an extra layer of resilience in many ways (from fault tolerance to ownership of encryption keys).
- 2) the actual data is in just one place. If you'd swap the local filesystem for S3 you see why this has to work that way. Voldemort simply gives one pointer to the file location. If the underlying filesystem is not available on the web server under the same mount point things will fail. Same goes when testing multiple WebDAV instances.
- 3) there are multiple solutions to 2).

- use a shared filesystem between instances. From NFS upwards, anything goes.
- use S3
- write your own filesystem interface in Scala mimicking the S3 or local filesystem driver but doing exactly what you want.

The setup needs to be changed by using NFS shares.

Credits:

Jakub Peisar (jakub.peisar-at-cesnet.cz)