CC1 - Cracow Cloud One

Release 2.0

2013, CC1 Team

April 25, 2014

CONTENTS

CHAPTER

ONE

CC1 INSTALLATION PROCEDURE

1.1 Introduction

CC1 software has been equipped with automated installation procedure so that transformation of the owned computing infrastructure (used in classic way) into private cloud (which is far more effective) is quick and doesn't require detailed insight into Cloud Computing techniques. Current automatic installation procedure has been prepared for the following Linux distros:

- Debian,
- Ubuntu.

Standard DEB packages mechanism is utilized. In the future it's planned to prepare automatic installation procedure for RPM-package-based OS-es. It's worth noting that any supported OS system may be chosen to run on the hardware since for cloud users OS is transparent. However it is recommended to use Debian 7 *Wheezy*. CC1 cloud uses KVM (Kernel-based Virtual Machine) environment for virtualization.

1.1.1 System structure

CC1 architecture is showd on the below figure - *architecture*.

System consists of several modules:

Web-Interface (WI) intuitive interface with user and admin module.

Cloud Manager (CLM) global cloud controller.

Cluster Managers (CM) distributed cluster-specific controllers.

EC2 interface ec2 interface allowes system access via popular Amazon EC2 interface.

Each module is associated with its process. Processes communicate with each other via web protocols (HTTP mainly). Two main installation types may be mentioned:

- *Type I*: each module on single server (computing node needs to be installed separatelly);
- *Type II*: any arrangement of modules on multiple servers.

In case of installation from packages, CLM and CM databases are created on the same machine. It may be adjusted further via config file. EC2 installation shall be described in separate chapter.

WI, CLM and CM may be installed and run on virtual server.

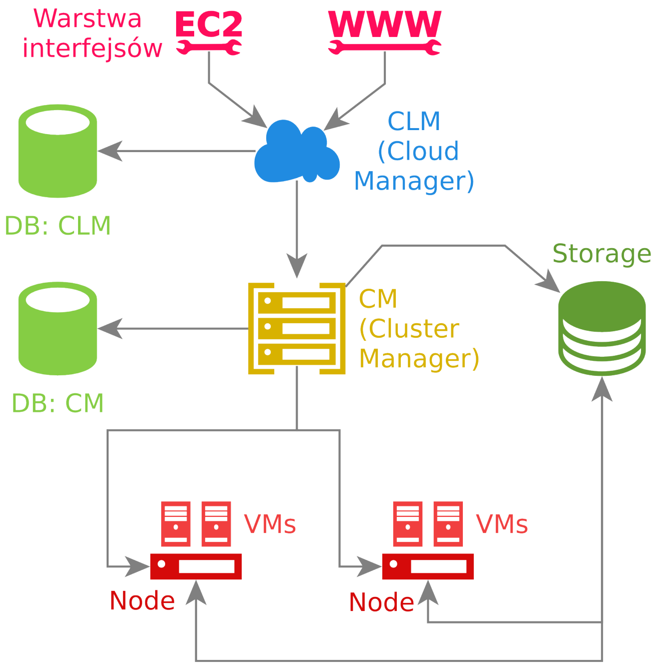

Figure 1.1: CC1 architecture

1.1.2 Installation choice

up to several thousands of cores	bigger deployments
Installation type I single multicore server hosting all the modules	Installation type II distributed servers instances for each module

In case of up-to-severa-thoudsends-cores deployments it's recommended to install all the management modules on a single server (*installation type I*).

Speedup of the system depends on the followig infrastructure elemets:

- disk array,
- network bandwidth,
- workstations efficiency.

CC1 controll processes do not impose any performance restrictions (even in case of huge deployments)

1.1.3 Requirements

Following elements are required for installation:

controll servers Servers to run module-related services are required. In case on single- machine installation (*type I*) it's recommended to use multicore server. Server's efficiency should be picked in proportion to the number of predicted users. Typical 4-core server with several GB of RAM is capable of serving few hundred registered users. Controll server may be run as multicore VM on powerful hardware.

worker nodes It's required to use at least one worker node. Worker nodes should be equipped with local storage with storage capacity proportional to the number of predicted VMs run simultanously on that node (before VM is ran its image is copied to the local storage of the specific worker node. To ensure the best functionality one ought to utilize minimal sizes of VMImages (enough to store operating system's data), since user's data may be stored on virtual storage disks (StorageImage). In case of Linux-based OSes this size doesn't exceed 10 GB.

disk array its capacity should be enough to store VM images and storage disks.

Currently automatic installation via DEB packages is available for Debian-based operating systems. CC1 packages are available in dedicated repository.

Warning: It is essential you configure your `locales` properly. Without that CM and CLM databases will not be configured and initialized. Issue the following and test for unset variables:

```
locales
```

In case some of those settings are empty one should reconfigure `locales` package, eg:

```
export LANGUAGE=en_US.UTF-8
export LANG=en_US.UTF-8
export LC_ALL=en_US.UTF-8
locale-gen en_US.UTF-8
dpkg-reconfigure locales
```

Caution: Unfortunately, for now it's required to ensure that if Apache2 is already installed and it's configured for HTTPs, it uses passwordless certificate. Otherwise installation procedure hangs (waiting for the password, which cannot be passed via postinstall script).

1.1.4 For those who update from CC1 v.1.7

Upgrade from v.1.7 is more time-consiming due to range of changes between mentioned versions.

It's highly recommended to create backup copy of the controll servers, databases and user resources on the disk array. Backups shall be stored on the system free of the critical operations (like creating new VMs). It implies the need to block user's access to the system.

Following modules need to be removed (by commands):

```
apt-get purge cc1-clm-v1.7
apt-get purge cc1-cm-v1.7
apt-get purge cc1-wi-v1.7
```

After version removal standard installation steps need to be executed. After that database migration is required.

1.2 Automatic installation - Type I: Single server

It's possible to run CC1 system within single server. Conf files are set for installation (*type I*) by default. Those files contain entries corresponding to netwok communication with other processes (`localhost` or `127.0.0.1`). Before installation one ought to:

1. Install pure Debian 7 (Wheezy) OS on servers meant to be controll server and virtualization nodes.

2. Configure network. Controll server and virtualization nodes should have access to Internet and belong to the same subset, where aren't any restrictions on *broadcast* packages.

Installations consists of two steps:

- controll modules installation,
- virtualization nodes installation.

After succesful installation one should prepare system to operate accordingly to *After installation* chapter.

1.2.1 CC1 controll modules: CLM, CM, WI

CC1 DEB package repository should be added to `/etc/apt/sources.list`. Next all the system modules should be installed:

```
echo "deb http://cc1.ifj.edu.pl/packages/ wheezy main" | tee -a /etc/apt/sources.list.d/cc1.list
apt-get update
apt-get install cc1-wi cc1-clm cc1-cm
```

`apt-get` tool suggests installation of the above packages altogether with their dependencies (e.g. libraries and required tools). One should confirm installation from untrusted source by entering `Y`. After succesful installation all the packages should be ready for use. In system new `cc1` user should appear:

```
cat /etc/passwd | grep cc1
```

CM and CLM databases

During installation process of the Cloud Manager (`cc1-clm`) and Cluster Manager (`cc1-cm`) default local databases (powered by **PostgreSQL**) are created and configured. Those work for local `cc1` user.

By default databases are filled with some initial data.

Note: If system is being updated from v1.7, former databases should be migrated.

CLM database migration (MySQL → PostgreSQL)

During CLM database's migration, Users and Groups data is migrated. Execute migration using following commands:

```
echo "delete from clm_user" | su cc1 -c "psql clm"
apt-get install python-mysqldb
python /usr/lib/cc1/manage.py clm mysql_migrate [OPTIONS]
```

Available options that point to v1.7's CLM database are as below:

```
--name=MYSQL_DB_NAME
  old database name, 'clm' by default

--password=MYSQL_USER_PASSWORD
  old database password, 'cc1' by default

--user=MYSQL_USER
  old database user, 'cc1' by default
```

```
--host=MYSQL_HOST
  old database host, '127.0.0.1' by default

--port=MYSQL_PORT
  old database port, 3306 by default
```

CM database migration (MySQL → PostgreSQL)

```
apt-get install python-mysqldb
python /usr/lib/cc1/manage.py cm mysql_migrate [OPTIONS]
```

Available options that point to v1.7's CM database are as below:

```
--name=MYSQL_DB_NAME
  old database name, 'cm' by default

--password=MYSQL_USER_PASSWORD
  old database password, 'cc1' by default

--user=MYSQL_USER
  old database user, 'cc1' by default

--host=MYSQL_HOST
  old database host, '127.0.0.1' by default

--port=MYSQL_PORT
  old database port, 3306 by default
```

1.3 Automatic installation - Type II: Distributed installation

Each controll module of the system (WI, CLM, CM) may be run on separated, dedicated server. All the virtualization nodes also need their own servers. For controll modules 3 (or less) servers with pure Debian OS shuld be prepared. There adequate processes shall be ran.

Note: One should remember to configure each CC1 module properly. By default configuration files are set for *installation type I*, where all services are ran on single machine. Distributed installation should be followed by edition of the specific values in `config.py` files of each controll module (`/etc/cc1/<moduł>/config.py`). One should especially care about proper IP's configuration. In case of TCP ports default settings may be kept.

Before CC1 packages installation one ought to pick IP numbers for each controll server. If one decides to make CC1 installation available via public IP number, additional public IP numbers are required for Web Interface, Cluster Manager and (optionally) EC2 Interface.

Communication between specific system modules is as below:

Todo

Example system structure

1.3.1 ☁ CLM controll server

Each CC1 Cloud's deployment requires `cc1-clm` package's installation.

Cloud Manager's (CLM) installation

One should add CC1 repository to sources list and install CLM module alongside with all its dependencies:

```
echo "deb http://cc1.ifj.edu.pl/packages/ wheezy main" | tee -a /etc/apt/sources.list.d/cc1.list
apt-get update
apt-get install cc1-clm
```

Cloud Manager's configuraiton

CLM default configuration may be altered via `/etc/cc1/clm/config.py` file edition. Settings to change are (e.g.):

- outbox server (for user notifications),
- CC1 user activation,
- CLM database.

CLM database

Note: If CLM is expected to utilise default local PostgreSQL database, this chapter may be ommited. Such a database is created automatically during system installation.

During Cloud Manager intallation, its default database is created. By default it's **PostgreSQL** local database that works for local `cc1` user. If remote database should be used, one may alter db connection parameters by editing `DATABASES` setting in `/etc/cc1/clm/config.py` file. Specified database should be properly configured for cooperation with CLM:

- user created (consistent with `USER` specified in CLM `config.py`),
- `clm` database created,
- privileges on `clm` database granted to that user,
- connection from CLM server to `clm` database enabled.

Example PostgreSQL configuration:

1. Create `cc1` user:

   ```
   CREATE USER cc1 WITH PASSWORD 'cc1'
   ```

2. Create `clm` database:

   ```
   CREATE DATABASE clm
   ```

3. Listen on connection from outside by PostgreSQL - set `listen_address` and `port` in `postgresql.conf` file (eg. `/etc/postgresql/9.1/main.postgresql.conf`)
4. Set privileges on `clm` database to `cc1` user - add following line to `pg_hba.conf` (eg. `/etc/postgresql/9.1/main/pg_hba.conf`):

   ```
   host clm cc1 <CLM_ADDRESS>/0 password
   ```

When database is ready, its structure should be created and migrated by following commands:

```
cc1_manage_db clm syncdb
cc1_manage_db clm migrate
```

Caution: Database should get initialized. One should follow diffrent steps depending on whether update from CC1 v1.7 or fresh install is performed.

CLM database initialization in case of fresh installation If fresh installation is performed (not upgrade from v1.7), one should initialize database:

```
cc1_manage_db clm loaddata /usr/lib/cc1/clm/initial_data.json
```

CLM database migration in case of update from CC1 v1.7 If one updates from CC1 v1.7 and wants to keep all the data, it's required to migrate CLM database (MySQL→PostgreSQL)

During CLM database's migration, Users and Groups data is migrated. Execute migration using following commands:

```
echo "delete from clm_user" | su cc1 -c "psql clm"
apt-get install python-mysqldb
python /usr/lib/cc1/manage.py clm mysql_migrate [OPTIONS]
```

Available options that point to v1.7's CLM database are as below:

```
--name=MYSQL_DB_NAME
  old database name, 'clm' by default

--password=MYSQL_USER_PASSWORD
  old database password, 'cc1' by default

--user=MYSQL_USER
  old database user, 'cc1' by default

--host=MYSQL_HOST
  old database host, '127.0.0.1' by default

--port=MYSQL_PORT
  old database port, 3306 by default
```

CLM admin panel access via Web Interface Right after all packages installation it's possible to login to default admin panel:

Login **cc1**

Haslo **cc1**

Adres **http://<cloud-address>/admin_clm**

1.3.2 CM controll server

For cloud to operate at least single Cluster Manager (CM) should be used. It's resources controll module. It also manages VMImages.

Cluster Manager's (CM) installation

CC1 repository should be added to sources list and CM module alongside with all dependencies should be installed:

```
echo "deb http://cc1.ifj.edu.pl/packages/ wheezy main" | tee -a /etc/apt/sources.list.d/cc1.list
apt-get update
apt-get install cc1-cm
```

Cluster Manager configuration (optional)

CM default configuration may be altered via `/etc/cc1/cm/config.py` file edition. Settings to change are (e.g.):

- VM monitoring,
- connection to Cloud Manager,
- email server,
- CM database.

CM database

Note: If CM is supposed to utilise default local PostgreSQL database, this chapter may be ommited. Such a database is created automatically during system installation.

Not only Cloud Manager, but also CM install its own PostgreSQL database. If remote database should be used, one may alter db connection parameters by editing `DATABASES` setting in `/etc/cc1/cm/config.py` file. Specified database should be properly configured for cooperation with CM:

```
cc1_manage_db cm syncdb
cc1_manage_db cm migrate
```

Caution: Database needs to be properly initialized. Steps differ for CC1 v1.7 update and for fresh install.

CM database initialization in case of fresh installation If fresh installation is performed (not upgrade from v1.7), one should initialize database:

```
cc1_manage_db cm loaddata /usr/lib/cc1/cm/initial_data.json
```

CM database migration in case of update from CC1 v1.7 If one updates from CC1 v1.7 and wants to keep all the data, it's required to migrate CM database (MySQL→PostgreSQL)

```
apt-get install python-mysqldb
python /usr/lib/cc1/manage.py cm mysql_migrate [OPTIONS]
```

Available options that point to v1.7's CM database are as below:

```
--name=MYSQL_DB_NAME
  old database name, 'cm' by default

--password=MYSQL_USER_PASSWORD
  old database password, 'cc1' by default

--user=MYSQL_USER
  old database user, 'cc1' by default

--host=MYSQL_HOST
```

```
  old database host, '127.0.0.1' by default

--port=MYSQL_PORT
  old database port, 3306 by default
```

1.3.3 WI Web Interface

One should add CC1 repository to sources list and install WI module alongside with all its dependencies:

```
echo "deb http://cc1.ifj.edu.pl/packages/ wheezy main" | tee -a /etc/apt/sources.list.d/cc1.list
apt-get update
apt-get install cc1-wi
```

Next `/etc/cc1/wi/config.py` configuration file should be edited.

1.4 After installation

After succesful installation of the controll server(s) of the CC1 system specific modules (WI, CLM, CM) may communicate with each other. One should login to CLM admin panel via WWW browser:

Address **http://<cloud-address>/admin_clm**

Login **cc1**

Password **cc1**

where `<cloud-address>` is the domain name or IP address of the server where WI is installed.

To gain full functionality of the system, one shoud execute following steps:

- *Register CM in CLM*
- *Register storage space in CM*
- *Register nodes in CM*
- *Prepare system to operate*

1.4.1 Register CM in CLM

From CLM admin panel chose **Cluster Managers**, where empty list should be displayed. After pressing **+ Add new CM** button at the bottom, simple form should be displayed, where new CM may be registered. Form fields:

Name Allowed: lower case letters, numbers and - dash.

Address Private IP address where CM module is ran. In case of installation *type I* `127.0.0.1` should be entered.

Port Default CM port number is 8001

Password Last two fields should contain CM admin password. The user who fills this form becomes newly-created CM's administrator automatically. It may be either main CLM administrator or any other user, who gained CLM privileges.

CM admin has access to CM admin panel under via following address: `http://<cloud-address>/admin_cm`

1.4.2 Register storage space in CM

Each CM shoud have its disk space, where user's disks shall be stored. This space should contain at least of single disk array. From the CM admin panel one should select *Storages* from *Hardware* menu entry. One may find **+ Create new Storage** button at the bottom. After clicking it approperiate form is displayed.

Name Disk array name should contain letters, numbers and **-** dashes.

Maximum capacity [MB] It's the maximal storage space on the disk array intended to be used by CC1

Address Here IP address of the disk array should be given. by default NFS is used as access protocol.

Directory Name of the directory provided by disk array. Its name will be used by `mount` command:

```
mount <nfs_server>:/<remote_dir> <local_dir>
```

Note: To avoid problems related to unsuccessful resources mounting, one should perform manual test on CM server and on any of the nodes. NFS client's installation needs to be confirmed. Next mount command should be executed. In case of NFS4 such a command goes as follows:

```
mount -t nfs4 <nfs_server>:/<remote_dir> /mnt
```

Mount should be checked for failure. It mast be ensured that *cc1* user has right to write. In single node *cc1* user is not created yet. It's only required to test mounting success. Next resource should be umounted.

1.4.3 Register nodes in CM

Next step is to add virtualization nodes to Cloud. There virtual machines will be ran. It's recommended to configure nodes automatically using commands available on CM. Computer dedicated to serve as Node should access internet and share network with CM and storage array. Its importat that there are no restrictions on broadcast packages within that network.

In order to install Node software automatically, one should login to CM server, where `cc1_cm_setup_node` tool is available. It allows automatic repository insert on Node, installation of specific packages and their proper configuration. Tool is also capable of adding new Node to Cluster Manager's database.

To configure , fire on CM server following commands:

```
cc1_cm_setup_node install <libvirt_url>
cc1_cm_setup_node configure <libvirt_url> <network_interface>
cc1_cm_setup_node add <libvirt_url> <cpu_total> <memory_total> <hdd_total>
```

where:

libvirt_url libvirt connection string is as follows:

```
qemu+ssh://cc1@<ip_noda>/system
```

`node_ip` is the IP number of the Node to configure. It may differ depending on used IP and used hypervisor. Username should always be `cc1`, its change may result in unexpected Cluster Manager's behaviour.

network_interface Usually `eth0` or comma-separated (`,`) list of network interfaces used for nodes communication. At least one needs to be provided. In case of redundant networks all the access interfaces may be provided.

cpu_total, memory_total, hdd_total *MB* in case of `memory` and `hdd_total`, *cores* in case of `cpu_total`. Describes amount of resources intended to be used by cloud. The rest of the resources may be used by OS.

Above tools also performs several tests designed to validate installation's correctness. After such an operation Node should be already configured, added to database and ready to operate. It's state may be checked in **Nodes** section within Web Interface admin panel.

```
http://<cloud_address>/admin_cm/
```

1.4.4 Prepare system to operate

Before first user accessses system, several administrative operations should be performed. The default admin is *cc1* user. It's created during installation process (it's default password is *cc1* - that password should be changed right after first login to system).

Enter CLM admin panel via Web Interface (`https://<cloud-address>/admin_clm`) and ...

- **define VM hardware templates** - enter *Templates→Create template*, fill form. It's recommended to use names that provide information about cores number and RAM amount, eg. *1 CPU 2 GB RAM*.
- **define range of the private IP addresses for Virtual Machines** - enter *Networks→Available pools*, add network for virtual machines. Such a pool must be consistent with one defined in Quagga configutaion file.
- **add public IP addresses pool** - enter *Networks→Public IP* and select *Add public IP address*.
- **Upload public images** - such images should be prepared and uploaded to Cluster Manager. It's recommended to install contextualization package on public images. Recommended image's side should not exceed 10 GB. Higher size will increase boot time of virtual machines (VM image is copied to local Node's disk).

  CTX installation

  – download installation script and run it:

  ```
  wget http://cc1.ifj.edu.pl/vmm/install.sh
  bash install.sh
  ```

  – check whether `cc1-vmm` service is present

One should select user registration method. CC1 system provides three various options:

- Email confirmation and CLM admin acceptation required. It requires providing email server settings and setting `MAILER_ACTIVE` flag in CC1 configuration file (`/etc/cc1/clm/config.py`)
- No confirmation via email, but CLM admin acceptation required. This option is enabled by default.
- Autoactivation without email and without CLM admin acceptation required.

1.5 System updates

CC1 comes with software updates intended to bring additional functionalities or bug fixes. Update may be performed without stopping users virtual machines.

If CC1 v2.0 was installed using DEB packages system, its modules may also be updated automatically via command:

```
apt-get install <module-name>
```

Warning: User shoud not have access to system during update. All critical operations should have already finished before proceeding (eg. creating Virtual Machines).

To update modules following commands should be executed:

```
apt-get update
apt-get install cc1-clm
apt-get install cc1-cm
apt-get install cc1-wi
```

Configuration files `config.py` in modules directories `/etc/cc1/<module>` aren't modified. If any new fields have been added to cloud configuration, installer notifies about it and new fields are stored in `changes.py` files `/etc/cc1/<module>/changes.txt`. Admin should alter module's configuration file accordingly to new fields list. Also database creation and migration scripts are installed as dependencies. In case of databases' schema modification databases' tables will be altered.