Installation of IMS network core

In this tutorial, the different servers of the IMS core network (HSS, P-CSCF, ICSCF, S-CSCF) are installed on the same machine, without any specific network configuration specific network configuration (localhost).

Step 1: Prerequisites

- apt-get update
- apt-get upgrade

Hardware and network requirements:

- A working Linux machine (ubuntu12.04-desktop-i386 is recommended);
- About 100 to 150 MB of free space;
- Preferably 1GB of RAM;
- A working Internet access.

Tools and Libraries

Install the following libraries:

- For source retrieval: SVN (`subversion`)
- For compilation: GCC3/4, `make, ant`
- Lexical and syntactic analyzer: `flex, bison`
- JDK7: `openjdk-7-jdk`
- `libmysql++3, libmysql++-dev, libmysql-java, libxml2, libxml2-dev`
- `curl, libcurl4-gnutls-dev`
- `ipsec-tools` (pour le chiffrement de clés)
- `mysql-server`
- `bind9`
- `openssl`

Step 2: Get the OpenIMSCore source code (via SVN)

Installation directory: the source code is pre-configured to work with a standard file path.

- Create a folder `/opt/OpenIMSCore/` and go there:
  - `mkdir /opt/OpenIMSCore`
  - `cd /opt/OpenIMSCore`
- Create `ser_ims` folder and put the PCSCF, ICSCF and SCSCF servers in it:
  - `mkdir ser_ims`
  - `svn checkout https://svn.code.sf.net/p/openimscore/code/ser_ims/trunk ser_ims/`
- Create `FHoSS` folder and place the HSS server in it:
  - `mkdir FHoSS`
  - `svn checkout https://svn.code.sf.net/p/openimscore/code/FHoSS/trunk FHoSS/`

Step 3 : Compilation of openIMSCore

- Compiling. cscf servers (ser_ims):
    - `cd ser_ims`
    - `make install-libs all`

Note: If an error occurs during compilation, it is probably that a library is missing: studied the error, it should display the "missing library".

- Compiling the FHoSS server:

A JDK >=1.5 must be installed on the machine. To make sure of this:".

Warning: If you have just installed a JDK>=1.5, check that this last version is well version is taken into account by the system (it is possible that your system is still linked to an old still linked to an old version of the JDK, despite the installation of a more recent JDK version).

- Compilation and deployment:
    - `cd ../FHoSS`
    - `ant compile`
    - `ant deploy`

It's done!

Step 4: Environment configuration: DNS and MySQL

Precisions:

Remember: the initial configuration of openIMSCore only allows foroperation in localhost (at first anyway).

Also, the default domain is "open-ims.test". It is better not to change it for the moment.

MySQL configuration:

Go to the openIMS folder and perform the following 3 dumps:

- `cd /opt/OpenIMSCore/`
- `mysql -u root -p -h localhost < ser_ims/cfg/icscf.sql`
- `mysql -u root -p -h localhost < FHoSS/scripts/hss_db.sql`
- `mysql -u root -p -h localhost < FHoSS/scripts/userdata.sql`

The password requested is the MySQL superuser password.

Check that the database is accessible and loaded:

- `mysql -u root -p -h localhost`
- `show databases;`

You should see 2 more tables: `hss_db and icscf.`

DNS configuration:

A sample DNS zone file is present in the installation directory of installation directory. Copy it to your DNS configuration directory:

- `cp ser_ims/cfg/open-ims.dnszone /etc/bind/`

Edit the named.conf.local (always in /etc/bind/) to take into account this new zone (add the following lines new zone (add the following lines):

```
zone "open-ims.test" {
type master;
file "/etc/bind/open-ims.dnszone";
} ;
```

Check that the named.conf file calls this file. To do this, the following line must be present in named.conf:

- `include "/etc/bind/named.conf.local";`

Add it if it is not the case.

Edit the /etc/resolv.conf file, delete (or comment out) the existing content and add the following lines:

```
search open-ims.test
domain open-ims.test
nameserver 127.0.0.1
```

Warning: Each time the OS is restarted, this file will be overwritten by the Network Manager. So, make a backup of this file (we will use it in our automatic server launch script, which we will script, which we will see at the end of this tutorial):

- `cp /etc/resolv.conf /etc/resolv.conf.ims`

Edit the /etc/hosts file and add the following lines:

```
127.0.0.1 localhost
```

```
127.0.0.1 open-ims.test ue.open-ims.test presence.open-ims.test
icscf.openims.test scscf.open-ims.test pcscf.open-ims.test hss.open-
ims.test
```

Warning: As before, make a backup of this file:

- `cp /etc/hosts /etc/hosts.ims`

Restart the DNS server:

- `/etc/init.d/bind9 restart`

Check that the configuration change has been taken into account (via dig):

- `dig open-ims.test`

Step 5 : OpenIMS Server Configuration and Launch Files

Copy the .cfg and .xml configuration files, as well as the .sh scripts of the x-cscf servers in the directory of your choice (we have chosen to place them in the directory of your choice (we have chosen to place them in the openIMS directory for simplicity and organization):

- `cd /opt/OpenIMSCore/`
- `cp ser_ims/cfg/*.cfg /opt/OpenIMSCore/`
- `cp ser_ims/cfg/*.xml /opt/OpenIMSCore/`
- `cp ser_ims/cfg/*.sh /opt/OpenIMSCore/`

You can edit its files according to your preferences

Step 6: Launching openIMS

- x-cscf servers:

Run `pcscf.sh, icscf.sh and scscf.sh`. These three servers must run in parallel; you can run these three servers on the same VM. You can run these three servers on the same VM, this is not a technical problem (however, not so good in terms of performance)

Keep the terminals open, you should be able to observe all the traffic between these servers.

- FHoSS server:

Launch the server via `FHoSS/deploy/startup.sh`:

- `cd FHoSS/deploy`
- `./startup.sh`

It is possible that you get an error when launching the script; in this case, it is most likely related to the environment variable JAVA_HOME. Check that this variable has been correctly exported. If this is not the case, run the following command in a terminal:

`export JAVA_HOME="/usr/lib/jvm/java-1.7.0-openjdk-i386"`

Modify the path according to your tree structure. Moreover, it is also possible that this variable is overwritten at each reboot of the machine.

Check that the FHoSS is correctly launched by accessing the interface

`http://localhost:8080/`

`- Login: hssAdmin`

`- Password: hss`

Step 7: Creating an auto start script

As explained earlier in this tutorial, it can be useful to create a script that automatically launches the different servers and that takes care of the necessary configurations before launching the servers (for example update/export the JAVA_HOME variable, etc.).
You can create a file with sh (example: `startUpOpenIMS.sh`) extension and store these instructions:


```
#!/bin/bash

echo "-- Files configuration --"
```

```
echo ""

echo "-- Writting in resolv.conf"

cp /etc/resolv.conf.ims /etc/resolv.conf

echo "-- End of writting in resolv.conf"

echo ""

echo "-- Writting in hosts --"

cp /etc/hosts.ims /etc/hosts

echo "End of writting in hosts --"

echo ""

echo"-- Restart bind9 server --"

/etc/init.d/bind9 restart

export JAVA_HOME="/usr/lib/jvm/java-1.7.0-openjdk-i386"

export PATH=$JAVA_HOME/bin:$PATH

mkdir /var/run/opensips

chmod 777 /var/run/opensips

echo""

echo"-- Starting Server x-cscf and FHoSS"

gnome-terminal \

--tab -e "bash -c 'cd /opt/OpenIMSCore && sudo ./pcscf.sh';bash" \

--tab -e "bash -c 'cd /opt/OpenIMSCore && sudo ./icscf.sh';bash" \

--tab -e "bash -c 'cd /opt/OpenIMSCore && sudo ./scscf.sh';bash" \

--tab -e "bash -c 'cd /opt/OpenIMSCore/FHoSS/deploy && ./startup.sh';bash"
```