

# OpenXCAP and OpenSIPS Presence Server

## Install Guide

The tutorial is a step by step install guide for setting up a OpenSIPS SIP presence server with privacy rules support using OpenXCAP as XCAP server. The explications are against Debian stable ( Etch) operating system.

## I. Install OpenXCAP

### 1. Prepare Installation

There are some packages that OpenXCAP depends on. Some are found in the Debian stable repository and others must be installed manually.

To install the ones present in the repository run command:

```
apt-get install python-mysqldb libxml2 python-lxml python-zopeinterface python-twisted  
python-twisted-web python-twisted-web2 libgnutls13
```

There are two more packages that are not found in the stable repository: **python-application** and **python-gnutls** that must be installed from debs and sources respectively. And another two packages, **python-twisted-web2** and **python-mysqldb** that should have a newer versions than the one in the repository.

## 1. Install python-application

The python-application package can be found in the Debian testing repository.

Run this command to download the package:

```
wget http://ftp.fi.debian.org/debian/pool/main/p/python-application/python-application_1.0.9-4_all.deb
```

( this package as well as all the other referenced in this tutorial can be found in the **src** directory of this archive)

The package is not fully compatible with the stable Debian release. It requires a version of **python-support** newer than the one in the stable repository. You should download this from the link:

```
wget http://debian.linux-m32r.org/pool/main/p/python-support/python-support_0.7.7_all.deb
```

Run the following command to install python-application:

```
dpkg -i python-support_0.7.7_all.deb  
dpkg -i python-application_1.0.9-4_all.deb
```

## 2. Install python-gnutls

The package is also present in the Debian testing repository but it can not be installed from there as it requires an update on libc. Therefore it has to be installed from sources.

Take the sources from here:

```
wget http://pypi.python.org/packages/source/p/python-gnutls/python-gnutls-1.1.5.tar.gz
```

This package depends on 3 other packages.

The first one is python-ctype, that is present in the stable repository, so install it with:

```
apt-get install python-ctypes
```

The other two are **libcrypt11** and **libgnutls26** that will be installed from sources.

First prepare your system by installing the following packages:

```
apt-get install gcc g++ make libc6-dev python-dev
```

## **2.a). Install libgcrypt11**

Download the sources

```
wget http://ftp.de.debian.org/debian/pool/main/libg/libgcrypt11/libgcrypt11_1.4.1.orig.tar.gz
```

Extract with

```
tar -xvf libgcrypt11_1.4.1.orig.tar.gz
```

Change directory

```
cd libgcrypt-1.4.1
```

It requires a library. Install it with:

```
apt-get install libgpg-error-dev
```

Then install libcrypt

```
./configure
```

```
make
```

```
make install
```

## **2.b). Install libgnutls26**

Download the sources

```
wget ftp://ftp.gnutls.org/pub/gnutls/gnutls-2.2.2.tar.bz2
```

Extract

```
apt-get install bzip2
```

```
bzip2 -d gnutls-2.2.2.tar.bz2
```

```
tar -xvf gnutls-2.2.2.tar
```

Change directory

**cd gnutls-2.2.2**

Install

**./configure**

**make**

**make install**

Make symbolic links

**ln -s /usr/local/lib/libgnutls.so.26 /usr/lib/libgnutls.so.26**

**ln -s /usr/local/lib/libgnutls-extra.so.26 /usr/lib/libgnutls-extra.so.26**

## **2.c).Install python-gnutls**

Go to the directory where you have the sources.

Extract

**tar -xvf python-gnutls-1.1.5.tar.gz**

Change directory

**cd python-gnutls-1.1.5**

Install

**python setup.py build**

**python setup.py install**

## **3. Install python-twisted-web2**

Download the sources

**wget http://ftp.de.debian.org/debian/pool/main/t/twisted-web2/twisted-web2\_8.1.0.orig.tar.gz**

Extract

```
tar -xvf twisted-web2_8.1.0.orig.tar.gz
```

Change Directory

```
cd TwistedWeb2-8.1.0
```

Install

```
python setup.py build
```

```
python setup.py install
```

#### **4. Install python-mysqldb**

Download the sources

```
wget http://ftp.de.debian.org/debian/pool/main/p/python-mysqldb/python-  
mysqldb_1.2.2.orig.tar.gz
```

Extract

```
tar -xvf python-mysqldb_1.2.2.orig.tar.gz
```

Change directory

```
cd MySQL-python-1.2.2
```

Install

```
python setup.py build
```

```
python setup.py install
```

## 5. Install OpenXCAP

Download the sources

```
wget http://download.ag-projects.com/OpenXCAP/openxcap_0.9.9.tar.gz
```

Extract

```
tar -xvf openxcap_0.9.9.tar.gz
```

Change directory

```
cd openxcap
```

Install

```
python setup.py build
```

```
python setup.py install
```

You will find in the archive in **config** directory a file named `openxcap_config.ini` . This is the configuration file for openxcap. You must copy it in `/etc/openxcap/config.ini`

```
mkdir /etc/openxcap
```

```
cp path_to_archive/openxcap_config.ini /etc/openxcap/config.ini
```

You need to edit this according to your configuration.

In the [Server] and [Authentification] sections change the 10.10.10.10 address to the address of the interface(IP or domain name) where the XCAP server will be listening.

In the [Database] and [OpenSER] sections change the database URL to the one used by the presence server. Configure the **xmlrpc\_url** in the [OpenSER] section to the address of the presence server and the port where the MI xmlrpc interface of the OpenSIPS presence server is listening ( configured in the 'port' parameter of the `mi_xmlrpc` module).

Copy the startup script, `debian/openxcap.init` to `/etc/init.d/openxcap`

```
cp debian/openxcap.init /etc/init.d/openxcap
```

```
chmod +x /etc/init.d/openxcap
```

Start the service:

```
/etc/init.d/openxcap start
```

You can find more information about OpenXCAP on the project site: [www.openxcap.org](http://www.openxcap.org)

## II. Install OpenSIPS

### 1. Get the sources

Download the the sources for the stable OpenSIPS release version 1.4.1 :

```
wget http://opensips.org/pub/opensips/latest/src/opensips-1.4.1-notls_src.tar.gz
```

### 2. Compile and Install

Extract:

```
tar -xvf opensips-1.4.1-notls_src.tar.gz
```

You can rename the directory to opensips\_presence: **mv opensips-1.4.1-notls opensips\_presence.**

Also it is recommended to move the sources in the default source directory: **mv opensips\_presence /usr/local/src/.**

Go to the directory

```
cd /usr/local/src/opensips_presence
```

Apart from the modules that are compiled by default, **presence**, **presence\_xml** , **db\_mysql** and **mi\_xmlrpc** modules are also needed. Not to compile each manually, you can remove them from the **exclude\_modules** list defined in Makefile file before running make. Open the file at line 51 and remove the modules mentioned above from the list.

There are some packages that OpenSIPS needs and which you should install beforehand. Run this command:

```
apt-get install bison flex libmysqlclient15-dev libxml2-dev libxmlrpc-c3-dev
```

You can configure a location where to install the server by editing the Makefile.defs file and defining

the prefix script variable( line 209). Example:

**prefix=/usr/local/opensips\_presence**

If you don't configure the prefix, the default locations will be used ( /usr/local/lib/opensips/ for libraries, /usr/local/etc/opensips/ for configuration file, etc.)

Compile and install :

**make all**

**make install**

### 3. Database

The presence server requires access to a database having presence tables installed. To create a new database you have to run the command:

**opensipsdbctl create**

If you use a prefix, you will find this command in the sbin directory where opensips is installed.  
vim opensipsdbctl – DBENGINE=mysql.

### 4. Configuration file

**You can have the presence server integrated with the proxy or separate, running on a different machine or a different port and handling only presence requests. For the latter you need to configure the proxy such that it forwards all presence requests( PUBLISH and NOTIFY) to the presence server.**

You will find in the **config** directory of the archive a configuration file example opensips\_presence.m4 for a separate presence server together with the file with the local definitions local.m4.

You need to edit these files. In the opensips\_presence.m4, the **mpath** variable has to be set to the path of the module libraries( line 33). The value present there is for the prefix set as in the above example. The meaning of the variables present in local.m4 file is:

MY\_IP : - the IP of the interface where the presence server will be listening for requests

MY\_PORT - the port used by the presence server

SERVER\_IP - the IP address of the proxy

SERVER\_PORT - the port used by the proxy



DB\_HOST, DB\_USER, DB\_PASSWD, DB\_NAME – address, username, password and name to be used for database access

Be sure to configure the **port** parameter of **mi\_xmlrpc** module to the same value as the port configured for the opensips server in the openxcap configuration file.

## 5. Starting script

To manipulate the server easily it is indicated to write a init script in the /etc/init.d/ directory. You will find a file example named opensips\_presence in the archive. Edit as needed and copy it in the /etc/init.d directory.

## 6. Start the server

Run:

```
/etc/init.d/opensips_presence start
```

## 7. Debugging

To check if the server is running, search for the process in the process table:

```
ps aux | grep opensips
```

If no process is found look for errors in the log file. It is indicated to add a filter in syslog for local0 , configured as log facility for OpenSIPS and send the log messages to a separate file( ex: /var/log/opensips\_presence).

For more info read the documentation from the OpenSIPS site:

**<http://opensips.org/index.php?n=Resources.Documentation>.**