

BDNOG3, Dhaka

Introduction to SIP Express Router

Anowar Hasan Sabir,
Sumon kumar saha,
Al Faruq Ibna Nazim, Jahangir Hossain

Notes on Kamailio vs SER

Since version 4.0.0, Kamailio and SER integration was fully completed, there are no more duplicated modules and a single database schema. Default flavour is kamailio, setting this name to the application and database. SER flavour can be compiled using make command line options (practically, flavour system is now just about naming the application and related components).

Prerequisites

To be able to follow the guidelines from this document you need root access. The following packages are required before proceeding to the next steps.

git client: `apt-get install git-core` - it is recommended to have a recent version, if your Linux distro has an old version, you can download newer one from: <http://git-scm.com/>

gcc compiler: `apt-get install gcc`

flex - `apt-get install flex`

bison - `apt-get install bison`

libmysqlclient-dev - `apt-get install libmysqlclient-dev`

make - `apt-get install make`

if you want to enable more modules, some of them require extra libraries:

libssl - `apt-get install libssl-dev`

libcurl - `apt-get install libcurl4-openssl-dev`

libxml2 - `apt-get install libxml2-dev`

libpcre3 - `apt-get install libpcre3-dev`

Getting sources from GIT

First of all, you have to create a directory on the file system where the sources will be stored.

```
mkdir -p /usr/local/src/kamailio-4.1  
cd /usr/local/src/kamailio-4.1
```

Download the sources from GIT using the following commands.

```
git clone --depth 1 --no-single-branch git://git.sip-router.org/kamailio  
kamailio  
cd kamailio  
git checkout -b 4.1 origin/4.1
```

Note: if your git client version does not support `--no-single-branch` command line parameter, then just remove it.

Tuning Makefiles

The first step is to generate build config files.

```
make cfg
```

Next step is to enable the MySQL module. Edit modules.lst file:

```
nano -w modules.lst
```

Add db_mysql to the variable include_modules.

```
include_modules= db_mysql
```

Save the modules.lst and exit.

Alternative is to set 'include_modules' variable to specify what extra modules to be included for compilation when building Makefile cfg:

```
make include_modules="db_mysql dialplan" cfg
```

If you want to install everything in one directory (so you can delete all installed files at once), say /usr/local/kamailio-devel, then set PREFIX variable to the install path in 'make cfg ...' command:

```
make PREFIX="/usr/local/kamailio-devel" include_modules="db_mysql  
dialplan" cfg
```

Compile Kamailio

Once you added the mysql module to the list of enabled modules, you can compile Kamailio:

```
make all
```

You can get full compile flags output using:

```
make Q=0 all
```

Install Kamailio

When the compilation is ready, install Kamailio with the following command:

```
make install
```

Where all installed

The binaries and executable scripts were installed in:

```
/usr/local/sbin
```

These are:

kamailio - Kamailio SIP server

kamdbctl - script to create and manage the Databases

kamctl - script to manage and control Kamailio SIP server

sercmd - CLI - command line tool to interface with Kamailio SIP server

Create MySQL database

To create the MySQL database, you have to use the database setup script. First edit kamctlrc file to set the database server type:

```
nano -w /usr/local/etc/kamailio/kamctlrc
```

Locate DBENGINE variable and set it to MySQL:

```
DBENGINE=MYSQL
```

You can change other values in kamctlrc file, at least it is recommended to change the default passwords for the users to be created to connect to database.

Once you are done updating kamctlrc file, run the script to create the database used by Kamailio:

```
/usr/local/sbin/kamdbctl create
```


Edit configuration file

To fit your requirements for the VoIP platform, you have to edit the configuration file.

```
/usr/local/etc/kamailio/kamailio.cfg
```

Follow the instruction in the comments to enable usage of MySQL. Basically you have to add several lines at the top of config file, like:

```
#!define WITH_MYSQL  
#!define WITH_AUTH  
#!define WITH_USRLOCDB
```

The init.d script

The init.d script can be used to start/stop the Kamailio server in a nicer way. A sample of init.d script for Kamailio is provided at:

`/usr/local/src/kamailio-4.1/kamailio/pkg/kamailio/deb/debian/kamailio.init`

Just copy the init file into the `/etc/init.d/kamailio`. Then change the permissions:

```
cp /usr/local/src/kamailio-4.1/kamailio/pkg/kamailio/deb/debian/kamailio.  
init /etc/init.d/kamailio  
chmod 755 /etc/init.d/kamailio
```

then edit the file updating the `$DAEMON` and `$CFGFILE` values:

```
DAEMON=/usr/local/sbin/kamailio  
CFGFILE=/usr/local/etc/kamailio/kamailio.cfg
```

You need also setup a configuration file in the /etc/default/ directory. This file can be found at:

```
/usr/local/src/kamailio-4.1/kamailio/pkg/kamailio/deb/debian/kamailio.  
default
```

You need to rename the file to 'kamailio' after you've copied it. Then edit this file and set `RUN_KAMAILIO=yes`. Edit the other options at your convenience.

Create the directory for pid file:

```
mkdir -p /var/run/kamailio
```

Default setting is to run Kamailio as user "kamailio" and group "kamailio". For that you need to create the user:

```
adduser --quiet --system --group --disabled-password \  
--shell /bin/false --gecos "Kamailio" \  
--home /var/run/kamailio kamailio
```

set ownership to /var/run/kamailio

chown kamailio:kamailio /var/run/kamailio

Then you can start/stop Kamailio using the following commands:

/etc/init.d/kamailio start

/etc/init.d/kamailio stop

Ready to rock

Now everything is in place. You can start the VoIP service, creating new accounts and setting the phones.

A new account can be added using 'kamctl' tool via 'kamctl add <username> <password> <email>'. (or try without the email)

```
kamctl add 1001 testpasswd 1001@mysipserver.com
```