Short & Logical Howto on

Next Generation Network (NGN) Technology

By:

Onno W. Purbo

Focused on
- Asterisk
- OpenSER
- ENUM

References:

http://www.asterisk.org
http://www.openser.org

Compile Asterisk in Ubuntu

Prepare Ubuntu for compiling software

```
# apt-get install kernel-package libncurses5-dev fakeroot wget bzip2 g++
```

Get source code zaptel, asterisk, asterisk-addons, asterisk-sounds from Asterisk

http://www.asterisk.org

When needed, we can download also mpg123 from

http://www.mpg123.de/download.shtml
http://sourceforge.net/project/showfiles.php?group_id=135704

Copy all source code to /usr/local/src/

```
# cp zaptel-1.4.9.2.tar.gz /usr/local/src/
# cp asterisk-1.6.0-beta5.tar.gz /usr/local/src/
# cp asterisk-addons-1.6.0-beta2.tar.gz /usr/local/src/
# cp asterisk-sounds-1.2.1.tar.gz /usr/local/src/
# cp mpg123-1.3.1.tar.bz2 /usr/local/src/
```

Open the source code

```
# cd /usr/local/src
# tar zxvf zaptel-1.4.9.2.tar.gz
# tar zxvf asterisk-1.6.0-beta5.tar.gz
# tar zxvf asterisk-addons-1.6.0-beta2.tar.gz
# tar zxvf asterisk-sounds-1.2.1.tar.gz
# tar jxvf mpg123-1.3.1.tar.bz2
```

Compile MPG123

```
# cd /usr/local/src/mpg123-1.3.1
# ./configure
# make
# make install
```

Compile zaptel

```
#  cd /usr/local/src/zaptel-1.4.9.2
# ./configure
# make
# make install
```

Be prepared, zaptel will download firmware from the Internet. Zaptel is needed if we want to use conferencing facilities in Asterisk.

Compile asterisk

```
# cd /usr/local/src/asterisk-1.6.0-beta5
# ./configure
# make
# make install
# make samples
```

Compile asterisk addons

```
# cd /usr/local/src/asterisk-addons-1.6.0-beta2
# ./configure
# make
# make install
# make samples
```

Install asterisk sound

```
# cd /usr/local/src/asterisk-sounds-1.2.1
# make install
```

Finished on compiling and installing asterisk.

Minimal Configuration of Asterisk 1.6 in Ubuntu

Not much to be done for minimal configuration on Asterisk 1.6. We need to edit two (2) files,

> /etc/asterisk/sip.conf
> /etc/asterisk/extensions.conf

sip.conf configuration

Minimalist entry for each phone number,

```
[2000]
context=default
type=friend
username=2000
secret=123456
host=dynamic
dtmfmode=rfc2833
rfc2833compensate=yes
```

extensions.conf configuration

Example to enable all 20xx extensions to be able to call to each other,

```
exten => _20XX,1,Dial(SIP/${EXTEN},30,rt)
exten => _20XX,2,HangUp
```

Example to enable to call 662xxxxxxx using a PSTN gateway at extension 2031

```
exten => _662.,1,Dial(SIP/${EXTEN}@2031,60,rt)
```

Install OpenSER

Install openser

```
$ sudo apt-get install openser openser-mysql-module
[sudo] password for user:
```

Edit file /etc/default/openser, make sure

```
RUN_OPENSER=yes
```

Edit /etc/openser/openser.cfg, make sure

```
loadmodule "mysql.so"
loadmodule "auth.so"
loadmodule "auth_db.so"


# modparam("usrloc", "db_mode",   0)
modparam("usrloc", "db_mode", 2)
modparam("auth_db", "calculate_ha1", yes)
modparam("auth_db", "password_column", "password")
modparam("mi_fifo", "fifo_name", "/tmp/openser_fifo")
```

Do / create /tmp/openser_fifo

```
# touch /tmp/openser_fifo
```

Activate / create openser database

```
# openser_mysql create
MySQL password for root:
...
```

Restart openser

```
# /etc/init.d/openser restart
```

Check running openser using

```
# openserctl moni
```

If openser hang / restart failed, re-install openser,

```
# apt-get remove openser openser-mysql-module
# apt-get install openser openser-mysql-module
# /etc/init.d/openser restart
```

Use openserctl

Preparing openserctl by editing /etc/openser/openserctlrc. Make sure

```
## your SIP domain
SIP_DOMAIN=openser.org
## database type: MYSQL or PGSQL, by defaulte none is loaded
DBENGINE=MYSQL
## database host
DBHOST=localhost
## database name
DBNAME=openser
## database read/write user
DBRWUSER=openser
## database read only user
DBROUSER=openserro
## password for database read only user
DBROPW=openserro
## database super user
DBROOTUSER="root"
## type of aliases used: DB - database aliases; UL - usrloc aliases
## - default: none
ALIASES_TYPE="DB"
## control engine: FIFO or UNIXSOCK
## - default FIFO
CTLENGINE="FIFO"
## path to FIFO file
OSER_FIFO="/tmp/openser_fifo"

or

## path to FIFO file
# OSER_FIFO="/tmp/openser_fifo"

## check ACL names; default on (1); off (0)
# VERIFY_ACL=1
## ACL names - if VERIFY_ACL is set, only the ACL names from below list
## are accepted
# ACL_GROUPS="local ld int voicemail free-pstn"
## presence of serweb tables - default "no"
# HAS_SERWEB="yes"
## verbose - debug purposes - default '0'
# VERBOSE=1
## do (1) or don't (0) store plaintext passwords
## in the subscriber table - default '1'
# STORE_PLAINTEXT_PW=0
```

Do / create /tmp/openser_fifo

```
# touch /tmp/openser_fifo
```

Run / use openserctl

```
# openserctl
/usr/sbin/openserctl 1.2 - $Revision: 2135 $
Existing commands:
```

```
-- command 'start|stop|restart'
restart ........................... restart OpenSER
start ............................. start OpenSER
stop .............................. stop OpenSER

-- command 'acl' - manage access control lists (acl)
acl show [<username>] .............. show user membership
acl grant <username> <group> ....... grant user membership (*)
acl revoke <username> [<group>] .... grant user membership(s) (*)

-- command 'alias_db' - manage database aliases
alias_db show <alias> .............. show alias details
alias_db list <sip-id> ............. list aliases for uri
alias_db add <alias> <sip-id> ...... add an alias (*)
alias_db rm <alias> ................ remove an alias (*)
alias_db help ..................... help message
   - <alias> must be an AoR (username@domain)"
   - <sip-id> must be an AoR (username@domain)"

-- command 'avp' - manage AVPs
avp list [-T table] [-u <sip-id|uuid>]
    [-a attribute] [-v value] [-t type] ... list AVPs
avp add [-T table] <sip-id|uuid>
    <attribute> <type> <value> ............ add AVP (*)
avp rm [-T table]  [-u <sip-id|uuid>]
    [-a attribute] [-v value] [-t type] ... remove AVP (*)
avp help ................................. help message
   - -T - table name
   - -u - SIP id or unique id
   - -a - AVP name
   - -v - AVP value
   - -t - AVP name and type (0 (str:str), 1 (str:int),
                            2 (int:str), 3 (int:int))
   - <sip-id> must be an AoR (username@domain)
   - <uuid> must be a string but not AoR

-- command 'db' - database operations
db exec <query> ..................... execute SQL query
db roexec <roquery> ................. execute read-only SQL query
db run <id> ......................... execute SQL query from $id variable
db rorun <id> ....................... execute read-only SQL query from $id
variable
db show <tabel> .................... display table content

-- command 'lcr' - manage least cost routes (lcr)
         * lcr *
  * IP addresses must be entered in dotted quad format e.g. 1.2.3.4 *
  * <uri_scheme> and <transport> must be entered in integer or text,*
  * e.g. transport '2' is identical to transport 'tcp'.            *
  *    scheme: 1=sip, 2=sips;   transport: 1=udp, 2=tcp, 3=tls     *
  * Examples:  lcr addgw_grp usa 1                                 *
  *            lcr addgw level3 1.2.3.4 5080 sip tcp 1             *
  *            lcr addroute +1  1 1                               *
lcr
show ...............................................................
          ............. show routes, gateways and groups
lcr
```

```
reload ......................................................................
          ............. reload lcr gateways
lcr addgw_grp
<grp_name> ......................................................
          .............. add gateway group, autocreate grp_id
lcr addgw_grp <grp_name>
<grp_id> ...........................................
          ............... add gateway group with grp_id
lcr rmgw_grp
<grp_id> .......................................................
          ............... delete the gw_grp
lcr addgw <gw_name> <ip> <port> <scheme> <transport>
<grp_id> ...............
          .............. add a gateway
lcr addgw <gw_name> <ip> <port> <scheme> <transport> <grp_id>
<prefix> ......
          .............. add a gateway with prefix
lcr addgw <gw_name> <ip> <port> <scheme> <transport> <grp_id> <prefix>
<strip>
          .............. add a gateway with prefix and strip
lcr rmgw
<gw_name> ........................................................
          ............... delete a gateway
lcr addroute <prefix> <from> <grp_id>
<prio> ..................................
          .............. add a route ( use  to match anything in <from> )
lcr rmroute  <prefix> <from> <grp_id>
<prio> ..................................
          ............. delete a route

-- command 'rpid' - manage Remote-Party-ID (RPID)
rpid add <username> <rpid> ......... add rpid for a user (*)
rpid rm <username> ................. set rpid to NULL for a user (*)
rpid show <username> ............... show rpid of a user

-- command 'speeddial' - manage speed dials (short numbers)
speeddial show <speeddial-id> ....... show speeddial details
speeddial list <sip-id> ............. list speeddial for uri
speeddial add <sip-id> <sd-id> <new-uri> [<desc>] ...
          ........................... add a speedial (*)
speeddial rm <sip-id> <sd-id> ....... remove a speeddial (*)
speeddial help ...................... help message
   - <speeddial-id>, <sd-id> must be an AoR (username@domain)
   - <sip-id> must be an AoR (username@domain)
   - <new-uri> must be a SIP AoR (sip:username@domain)
   - <desc> a description for speeddial

-- command 'add|mail|passwd|rm' - manage subscribers
add <username> <password> <email> .. add a new subscriber (*)
passwd <username> <passwd> ......... change user's password (*)
rm <username> ...................... delete a user (*)
mail <username> .................... send an email to a user

-- command 'cisco_restart' - restart CISCO phone (NOTIFY)
cisco_restart <uri> ................ restart phone configured for <uri>

-- command 'online' - dump online users from memory
online ............................. display online users
```

```
-- command 'monitor' - show internal status
monitor ........................... show server's internal status

-- command 'ping' - ping a SIP URI (OPTIONS)
ping <uri> ........................ ping <uri> with SIP OPTIONS

-- command 'ul|alias' - manage user location or aliases
ul show [<username>]............... show in-RAM online users
ul show --brief................... show in-RAM online users in short
format
ul rm <username> [<contact URI>].... delete user's UsrLoc entries
ul add <username> <uri> ........... introduce a permanent UrLoc entry
ul add <username> <uri> <expires> .. introduce a temporary UrLoc entry

-- command 'fifo'
fifo ............................. send raw FIFO command
```

Add user,

```
# openserctl add 2000 123456 nomail@nomail.com
MySql password for user 'openser@localhost': (default openserrw)
-e new user '2000' added
```

See online user, example extension 2001,

```
# openserctl ul show 2001

-e database engine 'MYSQL' loaded
-e Control engine 'FIFO' loaded
-e entering fifo_cmd ul_show_contact location 2001@192.168.0.2
200 OK
Contact::
<sip:2001@192.168.0.92>;q=;expires=3413;flags=0x0;cflags=0x0;socket=<udp:
192.168.0.2:5060>;methods=0x163F;user_agent=<Grandstream BT120 1.0.8.17>
-e FIFO command was:
:ul_show_contact:openser_receiver_27106
location
2001@192.168.0.2
```

User IP address is explisit in sip:2001@192.168.0.92

OpenSER Routing

To redirect all call to 08* to PSTN Gateway located at 192.168.0.95:5061. Use,

```
if(uri=~"sip:08[0-9]+@.*")
{
    # forward ke PSTN gateway di 192.168.0.95:5061
    rewritehostport("192.168.0.95:5061");
    route(1);
}
route[1] {
        # send it out now; use stateful forwarding as it works reliably
        # even for UDP2TCP
        if (!t_relay()) {
                sl_reply_error();
        };
        exit;
}
```

PSTN gateway would be better to use a fixed IP address.

How to remove (strip) and add (prefix) in phone number

```
if (uri=~"^sip:00[1-9][0-9]*@example\.net") {
    # strip leading "00"
    # (change example.net to your domainname
    # or skip the stuff after the "@")
    strip(2);
    # (adjust, if your international prefix is something else than "00")
    prefix("+");
};
```

Call PSTN Using OpenSER

Assumed,

* ATA Located at IP address 192.168.0.3 port 5060.

```
# attempt handoff to PSTN
if (uri=~"^sip:9[0-9]*@mydomain.com") {
    log("Forwarding to PSTN\n");
    t_relay_to( "192.168.0.3", "5060");
    break;
    };
```

OpenSER with Asterisk Backend

We can combine several asterisk machine behind an OpenSER front end.

On the Asterisk side, edit /etc/asterisk/sip.conf, make sure,

```
[general]
allowguest=yes
allowexternalinvite=yes
```

On OpenSER side, prepare & edit /etc/openser/openser.cfg. Rewrite host and port to Asterisk host & port. For example, 600 will be redirected to asterisk server located at 192.168.0.3:5060

```
if (uri=~"sip:600@.*") {
        rewritehostport("192.168.0.3:5060");
        route(1);
}
route[1] {
        # send it out now; use stateful forwarding as it works reliably
        # even for UDP2TCP
        if (!t_relay()) {
                sl_reply_error();
        };
        exit;
}
```

ENUM Query in OpenSER

Basic configuration

- Prepared ENUM module in OpenSER configuration.
- Perpared ENUM routing tables in OpenSER configuration.

ENUM Quesry in OpenSER basically translate ENUM into URI. OpenSER can then process the call.

Prepare ENUM module, edit /etc/openser/openser.cfg, make sure,

```
loadmodule "enum.so"
```

Configure ENUM to check a default ENUM top level domain, such as,

```
modparam("enum", "domain_suffix", "e164.th.")
modparam("enum", "i_enum_suffix", "e164.th.")
```

Default domain to be searched is e164.arpa.

Example of ENUM Query is,

```
# Somewhere in the route[x] section:

# if you want to make ENUM work with numbers starting with "00",
# use the following to convert "00" it into a "+"

if (uri=~"^sip:00[1-9][0-9]*@example\.net") {
    # strip leading "00"
    # (change example.net to your domainname
    # or skip the stuff after the "@")
    strip(2);
    # (adjust, if your international prefix is something else than "00")
    prefix("+");
};

# check if request uri starts with an international phone
# number (+X.), if yes, try to ENUM resolve in e164.arpa.
# if no result, try in nrenum.net

if (uri=~"sip:\+[0-9]+@example\.net") {
    # (change example.net to your domainname
    # or skip the stuff after the "@")
    if ( !enum_query("e164.arpa.") ) {
        enum_query("nrenum.net.");
    };
};
```

SIPp to test OpenSER performance

References

> http://www.openser.org/docs/openser-performance-tests/

Install SIPp

> # apt-get install sipp

Assuming the OpenSER is running at 192.168.0.2. Perpare the following configuration for the server side,

```
# ----------- global configuration parameters -----------------------

debug=3             # debug level (cmd line: -ddddddddddd)
fork=yes
log_stderror=no     # (cmd line: -E)
children=8

disable_tcp=yes
disable_dns_blacklist=yes
disable_dns_failover=yes

# Uncomment these lines to enter debugging mode
#fork=no
#log_stderror=yes
#

# listen=udp:192.168.2.102:5070
listen=192.168.0.2:5060

# ------------------ module loading --------------------------------
#set module path
mpath="/usr/lib/openser/modules/"

loadmodule "sl.so"

# ------------------------  request routing logic -------------------
# main routing logic

route{
      sl_send_reply("200", "OK");
}
```

name the file for example cfg-test-uas.cfg.

Test the configuration file, to see any error,

> # openser -c -f /etc/openser/cfg-test-uas.cfg

If no error, run the server,

```
# openser -f /etc/openser/cfg-test-uas.cfg
```

Run the test from other client PC.

```
$ sipp -sn 192.168.0.2:5060 -m 200000 -r 10000 -d 1 -l 70
```

ENUM Server Implementation

ENUM is to map phone number in the Internet. ENUM is basically using a Domain Name System (DNS).

ENUM implementation is fairly simple,

- Install BIND DNS Server
- Edit BIND database / DNS entry, use NAPTR for ENUM.

ENUM Logic

PSTN Number	ENUM	SIP account
+6655566666600	0.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th	2000@sip.ait.ac.th
+6655566666601	1.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th	2001@sip.ait.ac.th
+6655566666602	2.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th	2002@sip.ait.ac.th
+6655566666687	7.8.6.6.6.6.6.6.6.5.5.5.6.6.e164.th	8007@sip.ait.ac.th

Mapping process is done in several stages,

- PSTN Number --> ENUM   by Asterisk (Softswitch)
- ENUM --> SIP Account   by BIND (DNS Server)
- SIP Account -> Call Process   by Asterisk (Softswitch)

ENUM Server Configuration

For example, we get an alocation for +6655566666600 up to +6655566666699, we need to edit /etc/bind/named.conf.local, for domain, 6.6.6.6.6.6.5.5.5.6.6.e164.th by adding an entry

```
zone "6.6.6.6.6.6.5.5.5.6.6.e164.th" IN {
        type master;
        file "/etc/bind/6.6.6.6.6.6.5.5.5.6.6.e164.th.db";
        allow-update { none; };
};
```

Enter all phone number into 6.6.6.6.6.6.5.5.5.6.6.e164.th.db located
in /etc/bind/6.6.6.6.6.6.5.5.5.6.6.e164.th.db. An example entry for 6.6.6.6.6.6.5.5.5.6.6.e164.th.db is as follows,

```
$TTL    86400
@    IN SOA  ns.ait.ac.th admin.ait.ac.th. (
                42              ; serial (d. adams)
                3H              ; refresh
                15M             ; retry
                1W              ; expiry
                1D )            ; minimum

        IN NS ns.ait.ac.th.
```

```
0.0    NAPTR 10 100 "u" "E2U+sip" "!^.*$!sip:2000@sip.ait.ac.th!" .
1.0    NAPTR 10 100 "u" "E2U+sip" "!^.*$!sip:2001@sip.ait.ac.th!" .
2.0    NAPTR 10 100 "u" "E2U+sip" "!^.*$!sip:2002@sip.ait.ac.th!" .
7.8    NAPTR 10 100 "u" "E2U+sip" "!^.*$!sip:8700@sip.ait.ac.th!" .
```

Meaning

+6655566666600	0.0.6.6.6.6.6.6.5.5.5.6.6.e164.th	2000@sip.ait.ac.id
+6655566666601	1.0.6.6.6.6.6.6.5.5.5.6.6.e164.th	2001@sip.ait.ac.id
+6655566666602	2.0.6.6.6.6.6.6.5.5.5.6.6.e164.th	2002@sip.ait.ac.id
+6655566666687	7.8.6.6.6.6.6.6.5.5.5.6.6.e164.th	8007@sip.ait.ac.id

Don't forget to restart DNS after editing the file,

```
# /etc/init.d/bind restart
```

Check ENUM Instalation

Make sure /etc/resolv.conf point to our ENUM DNS Server, such as

```
nameserver ip-address-ENUM-server
```

Test ENUM number,

```
# dig NAPTR 7.8.6.6.6.6.6.6.6.5.5.5.6.6.e164.th
```

or

```
# dig NAPTR 7.8.6.6.6.6.6.6.6.5.5.5.6.6.e164.th @ip-address-ENUM-server
```

Result example,

```
# dig NAPTR 0.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th

; <<>> DiG 9.4.1-P1 <<>> NAPTR 0.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15738
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;0.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th. IN  NAPTR

;; ANSWER SECTION:
0.0.6.6.6.6.6.6.6.5.5.5.6.6.e164.th. 86400 IN NAPTR 10 100 "u" "E2U+sip" "!^.*
$!sip:2000@sip.ait.ac.th!" .

;; AUTHORITY SECTION:
6.6.6.6.6.6.5.5.5.6.6.e164.th. 86400 IN NS      localhost.
```

```
;; ADDITIONAL SECTION:
localhost.              604800  IN      A       127.0.0.1

;; Query time: 0 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Mar  4 14:44:31 2008
;; MSG SIZE  rcvd: 146
```