How to use Homer capture agents with VoipNow

Applies to all VoipNow versions!

There are two ways in which you can send the SIP traffic to Homer: either by mirroring the swich port of your VoipNow server, or using Homer's Capture Agent. The latter is installed on the VoipNow server and sends HEP packets to the Capture Node. The Capture Node, which can be installed according to the instructions found here, must be configured to accept packets either from SIP mirroring or Capture Agents. This article explains both methods.

Step-by-step guide

Capture SIP traffic with port mirroring

This method does not require any work on your VoipNow server.

If the switch supports it, simply mirror the VoipNow port to the Capture Node port. Then edit the Kamailio config file to receive these packets.

Here is an example of the Kamailio configuration from a CaptureNode with the IP 10.150.20.85 configured to receive SIP packets from port mirroring.

Please note the commented HEP sections commented and the non-commented mirroring ones.

```
 #!KAMAILIO
#
# Example configuration file for a sipcapture node
#
####### Global Parameters definitions #########
#
# Please, make all your configuration changes here
#
# *** To enable extra stats
#     - define WITH_STATISTIC_METHOD_EXTRA
#     - define WITH_STATISTIC_INVITE_1XX
#!substdef "!HOMER_DB_USER!homer_user!g"
#!substdef "!HOMER_DB_PASSWORD!homer_password!g"
#!substdef "!HOMER_LISTEN_PROTO!udp!g"
#!substdef "!HOMER_LISTEN_IF!0.0.0.0!g"
#!substdef "!HOMER_LISTEN_PORT!9060!g"
####### Global Parameters #########
debug=1
log_stderror=no
memdbg=5
memlog=5
log_facility=LOG_LOCAL1
fork=yes
children=5
/* uncomment the next line to disable TCP (default on) */
disable_tcp=yes
/* IP and port for HEP capturing) */
#listen=HOMER_LISTEN_PROTO:HOMER_LISTEN_IF:HOMER_LISTEN_PORT
/* enable it only in mirroring scenario, not for HEP! */
 #!define SIPCAPTURE_MIRRORING_PORT
#Max loops
max_while_loops=100
loadmodule "pv.so"
loadmodule "db_mysql.so"
loadmodule "sipcapture.so"
loadmodule "textops.so"
loadmodule "rtimer.so"
loadmodule "xlog.so"
loadmodule "sqlops.so"
loadmodule "htable.so"
loadmodule "tm.so"
loadmodule "sl.so"
loadmodule "siputils.so"
loadmodule "exec.so"

modparam("htable", "htable", "a=>size=8;autoexpire=400")
modparam("htable", "htable", "b=>size=8;autoexpire=31")
# TODO: tune autoexpire setting for htable "c"
modparam("htable", "htable", "c=>size=8;autoexpire=31")
```

```
modparam("rtimer", "timer", "name=ta;interval=60;mode=1;")
modparam("rtimer", "exec", "timer=ta;route=TIMER_STATS")
modparam("sqlops","sqlcon","cb=>mysql://HOMER_DB_USER:HOMER_DB_PASSWORD@127.0.0.1/homer_statistic")


# ----- mi_fifo params -----
####### Routing Logic ########
modparam("sipcapture", "db_url", "mysql://HOMER_DB_USER:HOMER_DB_PASSWORD@127.0.0.1/homer_data")
modparam("sipcapture", "capture_on", 1)
modparam("sipcapture", "hep_capture_on", 1)
modparam("sipcapture", "insert_retries", 5)
modparam("sipcapture", "insert_retry_timeout", 10)
#!ifdef SIPCAPTURE_MIRRORING_PORT
/* IP to listen. Port/Portrange apply only on mirroring port capturing */
modparam("sipcapture", "raw_socket_listen", "10.150.20.85:5060-5080")
/* Name of interface to bind on raw socket */
modparam("sipcapture", "raw_interface", "eth0")
/* activate monitoring/mirroring port capturing */
modparam("sipcapture", "raw_moni_capture_on", 1)
/* children for raw socket */
modparam("sipcapture", "raw_sock_children", 4)
/* Linux only */
/* Promiscious mode RAW socket. Mirroring port. */
modparam("sipcapture", "promiscious_on", 1)
/* activate BPF */
modparam("sipcapture", "raw_moni_bpf_on", 1)
#endif
/* insert delayed */
#modparam("sipcapture", "db_insert_mode", 1)

#Stats time
stats.min = 5 desc "My stats TIME min"
```

The SIP routing logic section is omitted as it is common for both HEP and mirroring scenarios.

Capture SIP traffic with the Homer Capture Agent

Homer's Capture Agent allows you to capture SIP traffic from any Linux server. For up-to-date instructions, check Homer's installation guide.

1. Before you proceed, please install the following packages.

   ```
   yum install expat expat-devel libpcap libpcap libtool automake autoconf
   ```

2. Then get the source files from the GIT repo and install the Capture Agent.

   ```
   cd /usr/src
     git clone https://github.com/sipcapture/captagent.git captagent
     cd captagent/captagent
     ./build.sh
     ./configure
     make && make install
   ```

3. Check that the capture agent is running:

   ```
   # captagent -h
   usage: captagent <-vh> <-f config>
      -h  is help/usage
      -v  is version information
      -f  is the config file
      -D  is use specified pcap file instead of a device from the config
      -c  is checkout
      -d  is daemon mode
   ```

4. Now that the capture agent is operational, you need to configure it to send packets to the Capture Node.

Here is an example for a configuration file. The Capture Node is at 10.150.20.87, UDP port 9000. The file is located by default in `/usr/local/etc/captagent/captagent.xml`

```
 <?xml version="1.0"?>
<document type="captagent/xml">
        <configuration name="core.conf" description="CORE Settings">
          <settings>
            <param name="debug" value="3"/>
            <param name="daemon" value="true"/>
            <param name="syslog" value="false"/>
            <param name="pid_file" value="/var/run/captagent.pid"/>
            <param name="path" value="/usr/local/lib/captagent/modules"/>
          </settings>
        </configuration>
        <configuration name="modules.conf" description="Modules">
          <modules>
                <load module="core_hep"/>
                <load module="proto_uni"/>
                <load module="proto_rtcp"/>
                <load module="capt_cli"/>
          </modules>
        </configuration>
        <!-- CORE MODULES -->
        <configuration name="core_hep.conf" description="HEP Socket">
          <settings>
            <param name="version" value="3"/>
            <param name="capture-host" value="10.150.20.87"/>
            <param name="capture-port" value="9000"/>
            <param name="capture-proto" value="udp"/>
            <param name="capture-id" value="2001"/>
            <param name="capture-password" value="myHep"/>
            <param name="payload-compression" value="false" />
          </settings>
        </configuration>
        <!-- PROTOCOLS -->
        <configuration name="proto_uni.conf" description="UNI Proto Basic capture">
          <settings>
            <param name="port" value="5060"/>
            <!-- <param name="portrange" value="5060-5090"/> -->
            <!--
                use -D flag for pcap import
                use "any" for all interfaces in your system
            -->
            <param name="dev"  value="eth0"/>
            <param name="promisc" value="true"/>
            <!--
                comment it if you want to see all IPProto (tcp/udp)
            -->
            <param name="ip-proto" value="udp"/>
            <param name="proto-type"  value="sip"/>
            <param name="sip-parse"  value="true"/>
            <param name="rtcp-tracking"  value="true"/>
            <param name="reasm"  value="false"/>
            <param name="tcpdefrag"  value="false"/>
            <param name="debug" value ="false"/>
            <param name="buildin-reasm-filter" value="false"/>
            <!--
                <param name="expire-timer" value ="60"/>
                <param name="expire-rtcp" value ="120"/>
            -->
            <!-- <param name="filter" value="not src port 5099"/> -->
                <!-- <param name="vlan" value="false"/> -->
                <!--
                        ((ip[6:2] &amp;amp; 0x3fff != 0) - syntax for REASM packets
                         if capturing sip messages, you can filter by method
                         you can specify which method to NOT match with !
                         <param name="sip_method" value="INVITE"/>
                -->
          </settings>
```

```
        </configuration>
        <configuration name="proto_rtcp.conf" description="RTCP capture">
          <settings>
            <!-- <param name="portrange" value="5060-5090"/> -->
            <param name="dev"  value="eth0"/>
            <param name="promisc" value="true"/>
            <param name="debug" value ="false"/>
            <!-- <param name="rtcp-json" value="false"/> -->
            <!-- <param name="send-sdes" value="false"/> -->
            <!-- <param name="filter" value="and not src port 5099"/> -->
            <!-- <param name="vlan" value="false"/> -->
          </settings>
        </configuration>
        <!-- CLI  -->
        <configuration name="capt_cli.conf" description="CLI socket">
          <settings>
            <param name="cli-host" value="localhost"/>
            <param name="cli-port" value="8909"/>
            <param name="cli-password" value="12345"/>
          </settings>
        </configuration>
</document>
```

The actual config file might have a different format in future versions, so don't copy the entire file - use it strictly as an example.

The lines we're most interested in are the following:

```
#capture node IP
<param name="capture-host" value="10.150.20.87"/>
#capture node port
<param name="capture-port" value="9000"/>
#capture node protocol
<param name="capture-proto" value="udp"/>
#capture agent SIP port
<param name="port" value="5060"/>
#capture agent interface
<param name="dev"  value="eth0"/>
```

The config file of the Capture Node Kamailio looks like this:

```
#!KAMAILIO
#
# Example configuration file for a sipcapture node
#
####### Global Parameters definitions #########
#
# Please, make all your configuration changes here
#
# *** To enable extra stats
#     - define WITH_STATISTIC_METHOD_EXTRA
#     - define WITH_STATISTIC_INVITE_1XX
#!substdef "!HOMER_DB_USER!homer_user!g"
#!substdef "!HOMER_DB_PASSWORD!homer_password!g"
#!substdef "!HOMER_LISTEN_PROTO!udp!g"
#!substdef "!HOMER_LISTEN_IF!0.0.0.0!g"
#!substdef "!HOMER_LISTEN_PORT!9000!g"
####### Global Parameters #########
debug=1
log_stderror=no
memdbg=5
memlog=5
log_facility=LOG_LOCAL1
fork=yes
children=5
/* uncomment the next line to disable TCP (default on) */
disable_tcp=yes
/* IP and port for HEP capturing) */
listen=HOMER_LISTEN_PROTO:HOMER_LISTEN_IF:HOMER_LISTEN_PORT
#Max loops
max_while_loops=100
loadmodule "pv.so"
loadmodule "db_mysql.so"
loadmodule "sipcapture.so"
loadmodule "textops.so"
loadmodule "rtimer.so"
loadmodule "xlog.so"
loadmodule "sqlops.so"
loadmodule "htable.so"
loadmodule "tm.so"
loadmodule "sl.so"
loadmodule "siputils.so"
loadmodule "exec.so"

modparam("htable", "htable", "a=>size=8;autoexpire=400")
modparam("htable", "htable", "b=>size=8;autoexpire=31")
# TODO: tune autoexpire setting for htable "c"
modparam("htable", "htable", "c=>size=8;autoexpire=31")
modparam("rtimer", "timer", "name=ta;interval=60;mode=1;")
modparam("rtimer", "exec", "timer=ta;route=TIMER_STATS")
modparam("sqlops","sqlcon","cb=>mysql://HOMER_DB_USER:HOMER_DB_PASSWORD@127.0.0.1/homer_statistic")
# ----- mi_fifo params -----


####### Routing Logic ########
modparam("sipcapture", "db_url", "mysql://HOMER_DB_USER:HOMER_DB_PASSWORD@127.0.0.1/homer_data")
modparam("sipcapture", "capture_on", 1)
modparam("sipcapture", "hep_capture_on", 1)
modparam("sipcapture", "insert_retries", 5)
modparam("sipcapture", "insert_retry_timeout", 10)
#modparam("sipcapture", "capture_node", "homer01")
#Stats time
stats.min = 5 desc "My stats TIME min"
```

The lines we're interested in are the following:

```
#make capture node kamailio listen on any IP, udp port 9000
substdef "!HOMER_LISTEN_PROTO!udp!g"
substdef "!HOMER_LISTEN_IF!0.0.0.0!g"
substdef "!HOMER_LISTEN_PORT!9000!g"
```

5. Now you can start the Capture Agent in daemon mode.

```
captagent -d
```

It will start sending packets to 10.150.20.87 and, if configured correctly, you should already see them in Homer's web interface.

If you see errors like these when you start the captagent, it means that Kamailio on the capture node is not listening at the specified address or it is unreachable.

```
[ERR] core_hep.c:535 send error
[ERR] core_hep.c:535 send error
[ERR] core_hep.c:535 send error
```

Capture SIP traffic with sngrep

This might be the easiest way to send HEP packets to Homer.

1. Install the rpm appropriate for your Linux distribution: http://packages.irontec.com/
2. Start the program in the background.

```
sngrep port 5060 -H udp:10.150.20.87:9000 --no-interface -q 1<&- &
```

For more details on sngrep, check this KB article.

Related articles

- How to create a configuration template for a certain SIP device
- How to add a Local Agent to a Queue
- How to set up a SIP channel to interconnect with Skype forBusiness account
- Understanding SIP devices provisioning permissions
- How to use Request Logs