OpenSIPS

APRICOT2010 VoIP workshop

Intro to OpenSIPS

- http://www.opensips.org/
- Latest version is 1.6.1
- Its a opensource SIP Proxy (compliant with RFC3261 SIP protocol)
- Used for large-volume applications. OpenSIPS can handle tens of thousands of calls per second even on moderate hardware.
- Extremely fast to forward requests
- Flexible scripting language to change behaviour of OpenSIPs
- Useful resources
  - http://www.opensips.org/index.php?n=Resources.DocsCookbooks
  - http://www.opensips.org/Resources/Documentation#toc2

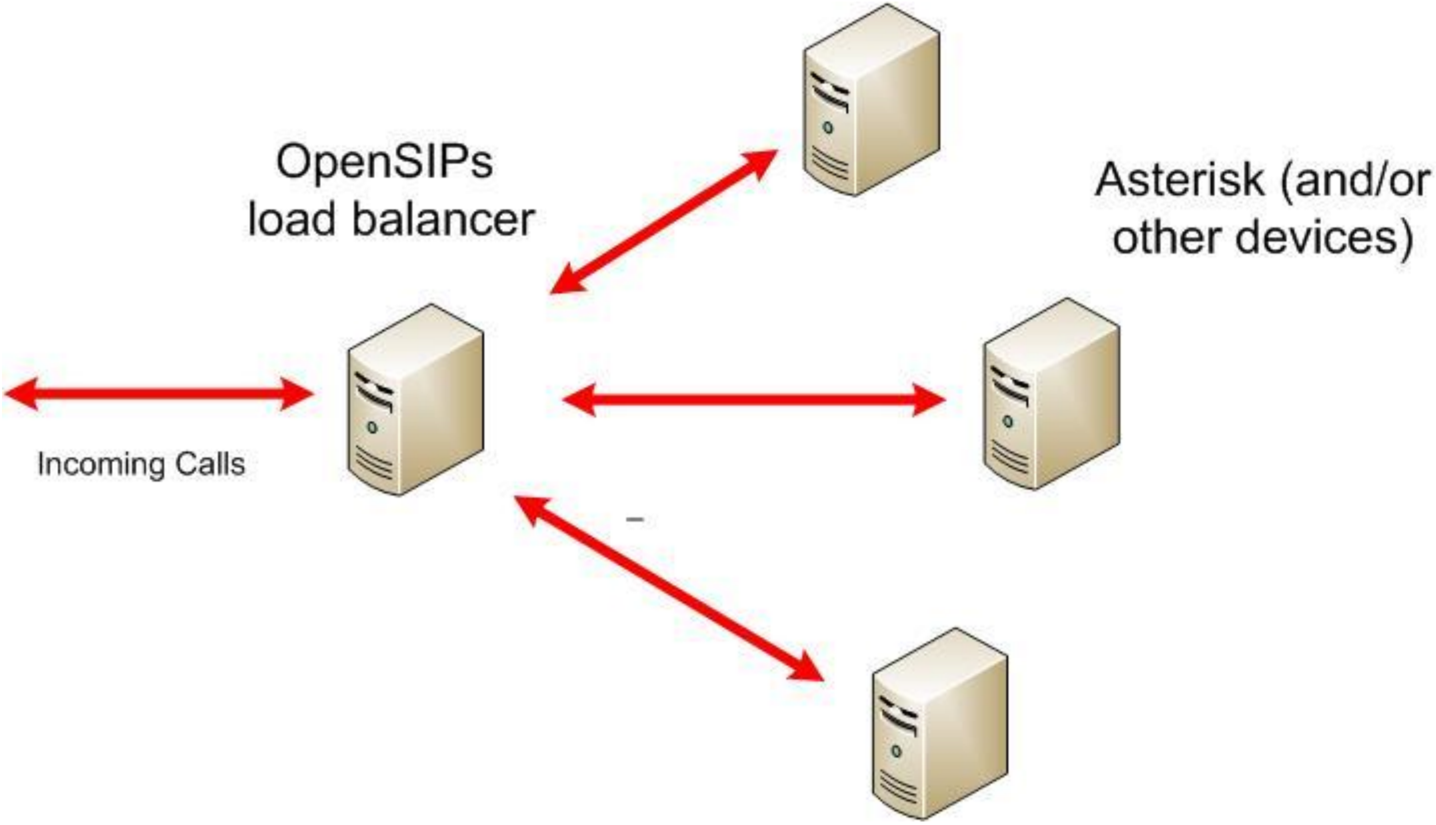
OpenSIPS history

- Originally based on SER (SIP Express Router)
- OpenSER was the first fork of SER in 2004
- OpenSER was renamed and forked in 2008
- Now there are two variants - Kamailio and OpenSIPs
- OpenSIPs seems to be the most active and dominant variant

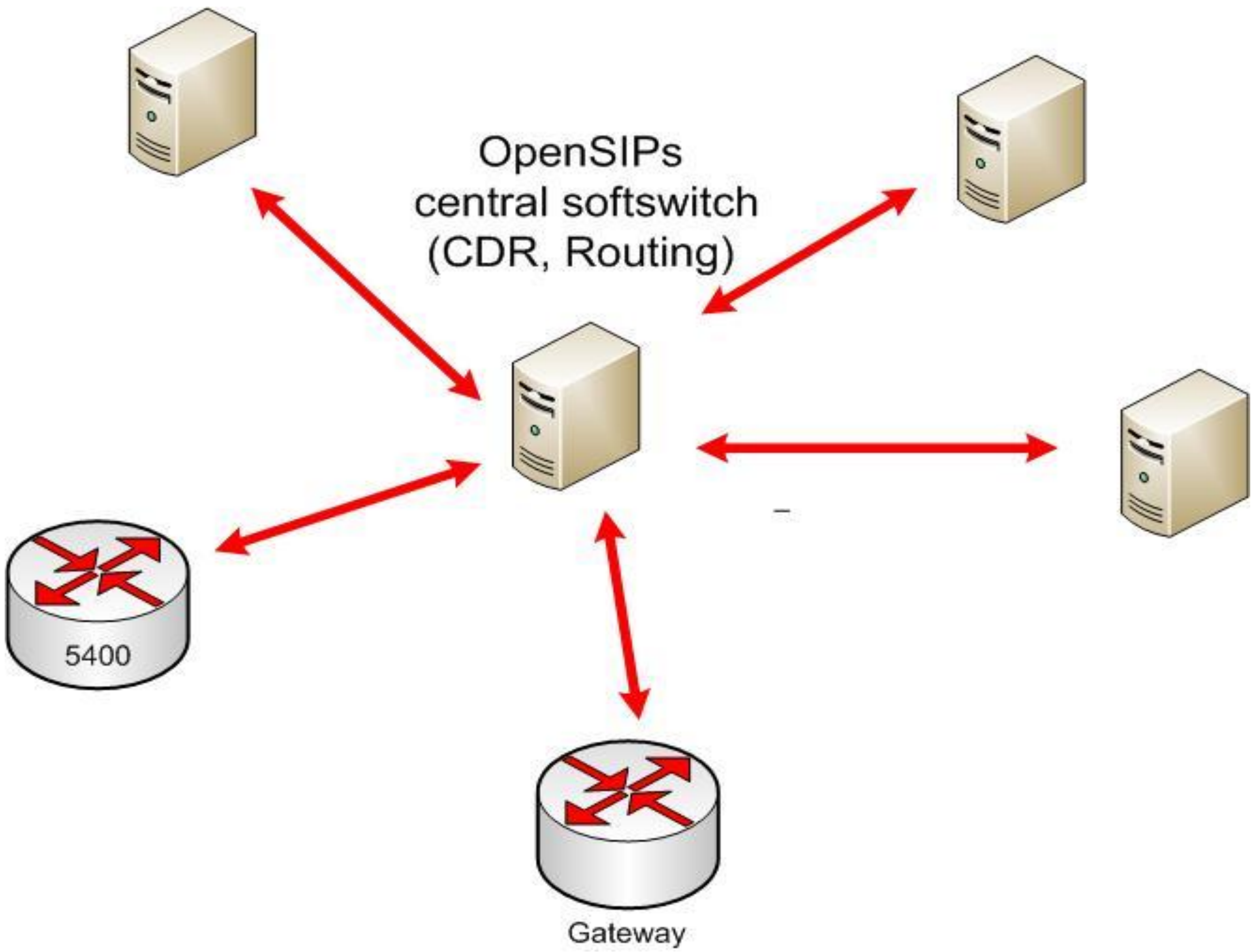
Uses of OpenSIPS

- SIP registrar server
- SIP router / proxy
- SIP redirect server
- SIP load-balancer or dispatcher
- SIP front end for gateways/asterisk
- Full VoIP provider solution and many other solutions

Load Balancing Architecture



Softswitch Architecture



OpenSIPS file locations

- Main configuration file: /etc/opensips/opensips.cfg

- Modules in /lib/opensips/modules. Where to look for missing modules
  - E.g. aaa_radius.so , load_balancer.so , ratelimit.so
- Binaries in /sbin/
  - opensips, opensipsctl, osipsconsole

- Log files in /var/log/messages
  - To see logs type at shell prompt: tail –f /var/log/messages

OpenSIPS main config

1) Global parameters

debug=3

log_stderror=no

log_facility=LOG_LOCAL0

2) Modules config to load modules

mpath="//lib/opensips/modules/”

loadmodule "signaling.so"

loadmodule "sl.so“

Etc

3) Module params

```
# ----- acc params -----
/* what sepcial events should be accounted ? */
modparam("acc", "early_media", 1)
modparam("acc", "report_ack", 1)
```

OpenSIPS main config

4) Routing logic

```
route{
    if (!mf_process_maxfwd_header("10")) {
        sl_send_reply("483","Too Many Hops");
        exit;
    }
    route(1);
}

# Route block
route[1] {
    if (is_method("INVITE")) {
        t_on_branch("2");
        t_on_reply("2");
        t_on_failure("1");
    }

    if (!t_relay()) {
        sl_reply_error();
    };
    exit;
}
```

OpenSIPS main config

4) Routing logic example

```
route{
    if (!mf_process_maxfwd_header("10")) {
        sl_send_reply("483","Too Many Hops");
        exit;
    }
    route(1);
}

# Route block
route[1] {
    if (is_method("INVITE")) {
        t_on_branch("2");
        t_on_reply("2");
        t_on_failure("1");
    }

    if (!t_relay()) {
        sl_reply_error();
    };
    exit;
}
```

OpenSIPS main config

Routing logic examples

```
route {
    # Respond to OPTIONS "PING"
    if( is_method("OPTIONS") )  {
        # send reply for each options request
        sl_send_reply("200", "ok");
        exit();
    }

    # Black list example
    if( uri=~"169\.223\.146\.100") {
        drop();
    }

    # Say OK to all REGISTER messages
    if (method=="REGISTER") {
        log("REGISTER");
        sl_send_reply("200", "ok");
        exit;
    };
}
```

OpenSIPS installation

- Download and compile source packages
  - cd /usr/src
  - wget http://opensips.org/pub/opensips/1.6.1/src/opensips-1.6.1-tls_src.tar.gz
  - tar -xzvf opensips-1.6.1-tls_src.tar.gz

- Compile and install modules
  - cd opensips-1.6.1-tls
  - make prefix=/ all
  - make prefix=/ install
  - mkdir /var/run/opensips

OpenSIPS startup

- **opensipsctl start|stop|restart**
- Startup options: **opensips -h**

```
version: opensips 1.6.1-tls (i386/linux)
Usage: opensips -l address [-l address ...] [options]
Options:
-f file Configuration file (default //etc/opensips/opensips.cfg)
-c Check configuration file for errors
-C Similar to '-c' but in addition checks the flags of exported
functions from included route blocks
-l address Listen on the specified address/interface (multiple -l
mean listening on more addresses). The address format is
[proto:]addr[:port], where proto=udp|tcp and
addr= host|ip_address|interface_name. E.g: -l locahost,
-l udp:127.0.0.1:5080, -l eth0:5062. The default behavior
is to listen on all the interfaces.
-n processes Number of child processes to fork per interface
(default: 8)
-r Use dns to check if is necessary to add a "received="
field to a via
```

OpenSIPS dispatcher module

- Used for load balancing across many devices
- Different load balancing algorithms. Round robin, hashing, random
- Stateless loadbalancing – doesn't gurantee fair distribution
- Put a list of destinations in a file or database to load balance across
- http://www.opensips.org/html/docs/modules/1.5.x/dispatcher.html#id271304

OpenSIPS dispatcher config

- Enable dispatcher module
  - `loadmodule "dispatcher.so"`

- Set dispatcher module parameters to read from dispatcher file
  - `modparam("dispatcher", "list_file", "/etc/dispatcher.list")`

- Routing logic

```
if (is_method("INVITE")) {
     ds_select_domain("1","4");
     t_relay();
 }
```

OpenSIPS dispatcher

- /etc/dispatcher.list

```
# gateways
1  sip:127.0.0.1:5060
1  sip:127.0.0.2:5060
1  sip:127.0.0.3:5060

# proxies
2  sip:127.0.0.4:5061
3  sip:127.0.0.8:5061
```

- Main dispatcher function
  - ds_select_domain(<set>,<algorithm>)

OpenSIPS Lab

1) Install OpenSIPS as per the slide notes above

2) Edit /etc/opensips/opensips.cfg and make sure it will start listening on port 5061

- mv /etc/opensips/opensips.cfg /etc/opensips/opensips.cfg.orig
- Use opensips.cfg located at http://169.223.146.13/lab/opensips.txt as your template and edit accordingly.

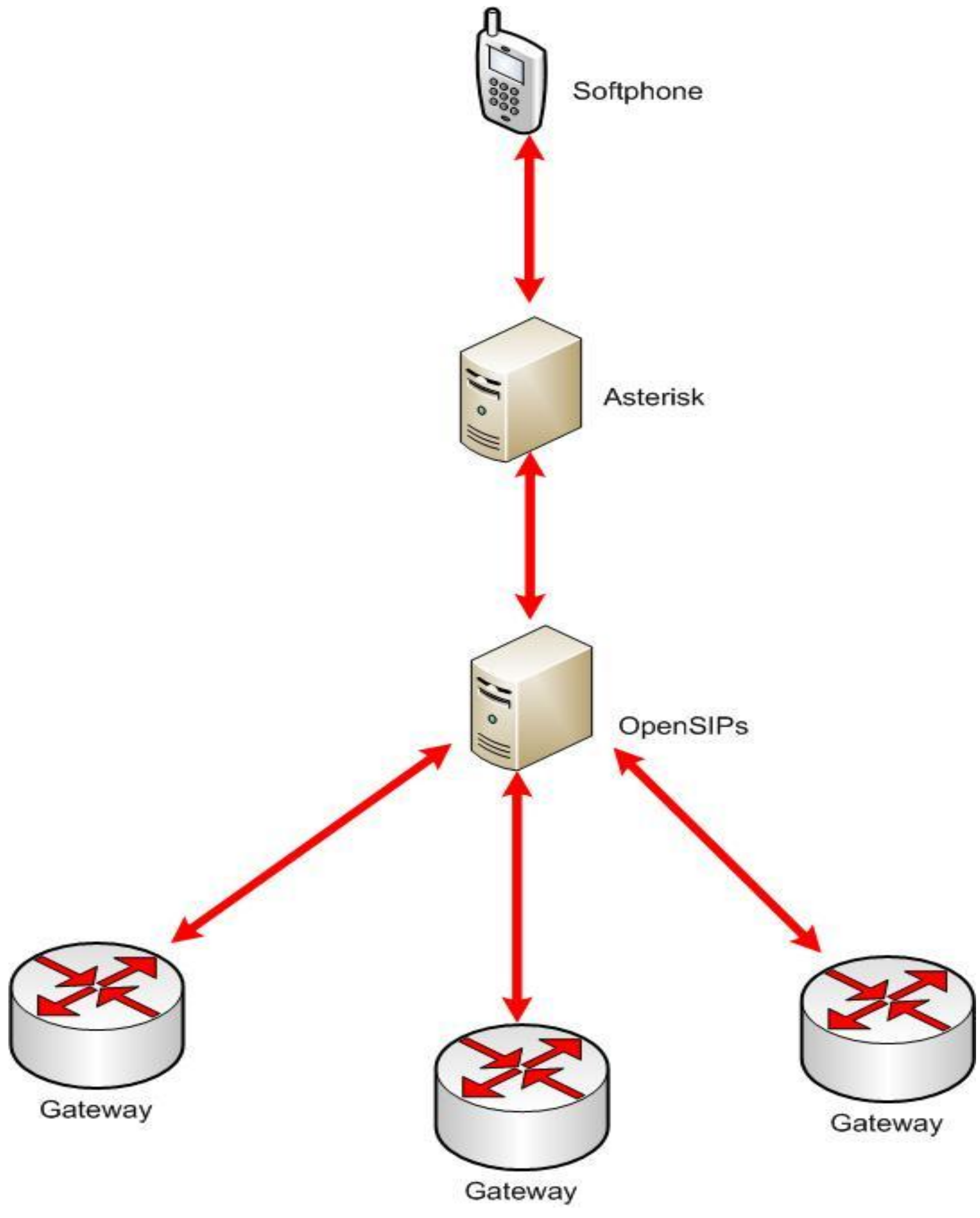
3) Configure a sip peer on asterisk to send calls to send to OpenSIPS (remember its listening on port 5061)

4) Configure extention 0084 in Asterisk to Dial to this new opensips peer

5) Configure dispatcher module in OpenSIPS to load balance calls across the three 2600 gateways.

- **Gateway 1 : 169.223.146.201**
- **Gateway 2: 169.223.146.202**
- **Gateway3: 169.223.146.203**

6) Make some calls and do a packet capture on your Asterisk server to see which gateway the calls are going to and the response codes


Softphone
Asterisk
OpenSIPs
Gateway
Gateway
Gateway