Firewall

About

If you are going to route traffic across your network, you need to ensure that you do not have a firewall blocking traffic on the specified network paths. Commonly firewalls are installed at egress points, where your local network connects to the internet. You may also run a firewall on the server itself. This is not always the case in some corporate environments. If you are not responsible for your network, you may wish to contact the group or individuals who are responsible for it.

Information they may need to enable traffic would include the port numbers. Various protocols use different ports for different things. Some you need bi-directional connectivity, some you only need outbound. Depending on your call scenario you may only need to allow connections that are initiated by your system. If you have any problems with protocols not working, it is recommended that you disable your firewall to the FreeSWITCH™ system, place a test call, and see if the problem persists. A misconfigured firewall is responsible for many common problems.

<table>
<thead>
<tr>
<th>Firewall Ports</th>
<th>Network Protocol</th>
<th>Application Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1719</td>
<td>UDP</td>
<td>H.323 Gatekeeper RAS port</td>
<td></td>
</tr>
<tr>
<td>1720</td>
<td>TCP</td>
<td>H.323 Call Signaling</td>
<td></td>
</tr>
<tr>
<td>2855-2856</td>
<td>TCP</td>
<td>MSRP</td>
<td>Used for call with messaging</td>
</tr>
<tr>
<td>3478</td>
<td>UDP</td>
<td>STUN service</td>
<td>Used for NAT traversal</td>
</tr>
<tr>
<td>3479</td>
<td>UDP</td>
<td>STUN service</td>
<td>Used for NAT traversal</td>
</tr>
<tr>
<td>5002</td>
<td>TCP</td>
<td>MLP protocol server</td>
<td></td>
</tr>
<tr>
<td>5003</td>
<td>UDP</td>
<td>Neighborhood service</td>
<td></td>
</tr>
<tr>
<td>5060</td>
<td>UDP &amp; TCP</td>
<td>SIP UAS</td>
<td>Used for SIP signaling (Standard SIP Port, for default Internal Profile)</td>
</tr>
<tr>
<td>5070</td>
<td>UDP &amp; TCP</td>
<td>SIP UAS</td>
<td>Used for SIP signaling (For default &quot;NAT&quot; Profile)</td>
</tr>
<tr>
<td>5080</td>
<td>UDP &amp; TCP</td>
<td>SIP UAS</td>
<td>Used for SIP signaling (For default &quot;External&quot; Profile)</td>
</tr>
<tr>
<td>8021</td>
<td>TCP</td>
<td>ESL</td>
<td>Used for mod_event_socket *</td>
</tr>
<tr>
<td>16384-32768</td>
<td>UDP</td>
<td>RTP/ RTCP multimedia streaming</td>
<td>Used for audio/video data in SIP, Verto, and other protocols</td>
</tr>
<tr>
<td>5066</td>
<td>TCP</td>
<td>Websocket</td>
<td>Used for WebRTC</td>
</tr>
<tr>
<td>7443</td>
<td>TCP</td>
<td>Websocket</td>
<td>Used for WebRTC</td>
</tr>
<tr>
<td>8081-8082</td>
<td>TCP</td>
<td>Websocket</td>
<td>Used for Verto</td>
</tr>
</tbody>
</table>

ESL SECURITY RISK
Think carefully about opening the ESL port to the external world and change the default password. ESL allows any system commands to be run or even to crash FreeSWITCH for call recovery testing. Allowing public access is therefore a security risk.

Note that the ports may vary depending on which modules you have loaded and their configuration, for instance you may have more or fewer SIP profiles, and you may have changed many of the above ports including SIP,RTP,ESL etc.

Linux netfilter iptables

THIS IS REQUIRED IF YOU ARE USING AN IPTABLES FIREWALL!!!
You must add the interface and port numbers for each sip_profile used in your FreeSWITCH installation.
The `nf_conntrack_sip` and `nf_conntrack_h323` modules will watch unencrypted SIP/H323 and automatically open the firewall ports required for RTP if you are accepting packets with the RELATED state. SIP and H323 packets after the first packet will be in the ESTABLISHED state. If you allow any RELATED, ESTABLISHED packets before processing new/unknown packets, then your firewall will accept subsequent packets much sooner, resulting in lower CPU usage and latency.

This may only apply to packets on the standard ports (UDP/5060, TCP/5060, TCP/1720) as it requires that the firewall recognizes the SIP/H323 protocol the packets are using.

This is of course not possible for encrypted connections, as the firewall cannot look inside the VoIP packets to get the RTP IPs and ports. With the imperative of encrypted communications, these steps are necessary to allow traffic through the firewall.

### Module options

You may need to adjust the following options in some configurations, e.g. setting sip_direct_media=0 for bypass media (either done by FreeSWITCH's bypass_media or by a gateway).

<table>
<thead>
<tr>
<th>Module</th>
<th>Option</th>
<th>Description</th>
<th>Values</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nf_conntrack_sip</code></td>
<td>sip_direct_signalling</td>
<td>Expect incoming calls from registrar only*</td>
<td>0/1</td>
<td>1</td>
</tr>
<tr>
<td><code>nf_conntrack_sip</code></td>
<td>sip_direct_media</td>
<td>Expect Media streams between signalling endpoints only</td>
<td>0/1</td>
<td>1</td>
</tr>
<tr>
<td><code>nf_conntrack_sip</code></td>
<td>sip_timeout</td>
<td>Timeout for the master SIP session</td>
<td>Integer</td>
<td>3600</td>
</tr>
<tr>
<td><code>nf_conntrack_sip</code></td>
<td>ports</td>
<td>Port numbers of SIP servers*</td>
<td>List of up to 8 port numbers (comma-separated) eg. 5060,5070,5080</td>
<td>5060</td>
</tr>
<tr>
<td><code>nf_conntrack_h323</code></td>
<td>default_rrq_ttl</td>
<td>Use this TTL if it's missing in RRQ</td>
<td>Integer</td>
<td>300</td>
</tr>
<tr>
<td><code>nf_conntrack_h323</code></td>
<td>gk routed_only</td>
<td>Only accept calls from gatekeeper</td>
<td>Integer</td>
<td>1</td>
</tr>
<tr>
<td><code>nf_conntrack_h323</code></td>
<td>callforward_filter</td>
<td>Only create call forwarding expectations if both endpoints are on different sides (determined by routing information)</td>
<td>Integer</td>
<td>1</td>
</tr>
</tbody>
</table>

Module options are given when loading the module, e.g. "modprobe nf_conntrack_sip ports=5060,5080" to use a non-default setting. Modules will need to be unloaded and reloaded to change their options.

The two most commonly used options are "ports" and "sip_direct_media".

You can check the current module parameter values by viewing the special files in `/sys/module/nf_conntrack_sip/parameters/`

### Configuration

If your distribution supports `/etc/modules.conf`:

```
/etc/modules.conf

open /etc/modules.conf in an editor
add the line:
options nf_conntrack_sip [options]
```

If your distribution supports `/etc/modprobe.d/`:

```
/etc/modprobe.d/

Create a file 50-nf_conntrack_sip.conf and open it in an editor
add the line:
options nf_conntrack_sip [options]
```

If your module is loaded using modprobe (eg. in rc.local):

```
```
modprobe
modprobe nf_conntrack_sip [options]

### Notes

sip_direct_signalling=1 and gkouted_only=1 will still allow all incoming calls if you have a rule such as 
```
p udp -dport 5060 :j ACCEPT
```
for the protocol, since you are explicitly allowing signaling instead of relying on the module.

Default values are taken from linux-source-2.6.26 on Debian 2010/02/26, and may vary in later versions.

### Side effects

Invalid SIP packets may be silently dropped by iptables even if all packets are allowed (e.g. if they have no CSeq header). This should not be an issue for FreeSWITCH, since it only sends valid SIP packets.

### Example

This is a basic firewall setup using just IPTABLES on Debian Squeeze.

**Back up your current iptables rules**

```
iptables-save > ~/iptables.up.rules.vanilla
```

**Create new rules**
Example iptables rules

vim ~/iptables.fs.rules

*mangle

# mark SIP UDP packets with CS3
-A OUTPUT -p udp -m udp --sport 5060 -j DSCP --set-dscp-class cs3
# mark SIP UDP packets with CS3
-A OUTPUT -p tcp --sport 5060 -j DSCP --set-dscp-class cs3
# mark SIP TLS packets with CS3
-A OUTPUT -p tcp --sport 5061 -j DSCP --set-dscp-class cs3
# mark RTP packets with EF
-A OUTPUT -p udp -m udp --sport 16384:32768 -j DSCP --set-dscp-class ef
COMMIT

*filter

# Allows all loopback (lo0) traffic
-A INPUT -i lo -j ACCEPT
# Drop all traffic to 127/8 that doesn't use lo0
-A INPUT ! -i lo -d 127.0.0.0/8 -j REJECT
# Accepts all established inbound connections
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
# Allow all outbound traffic
-A OUTPUT -j ACCEPT
# Allow SSH connections (THE -dport NUMBER IS THE SAME ONE YOU SET UP IN THE SSHD_CONFIG FILE)
-A INPUT -p tcp -m state --state NEW -dport 6245 -j ACCEPT
# Allow STUN service (Used for NAT traversal)
-A INPUT -p udp -m udp --dport 3478 -j ACCEPT
-A INPUT -p udp -m udp --dport 3479 -j ACCEPT
# Allow MLP protocol server
-A INPUT -p tcp -m tcp --dport 5002 -j ACCEPT
# Allow Neighborhood service
-A INPUT -p udp -m udp --dport 5003 -j ACCEPT
# Allow SIP UDP
-A INPUT -p udp -m udp --dport 5060 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 5060 -j ACCEPT
-A INPUT -p udp -m udp --dport 5061 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 5061 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 5060 -j ACCEPT
-A INPUT -p udp -m udp --dport 16384:32768 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8080 -s 192.168.0.122 -j ACCEPT
-A INPUT -p udp -m udp --dport 8080 -s 192.168.0.122 -j ACCEPT
-A INPUT -p udp -m udp --dport 8080 -s 192.168.0.122 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8080 -s 192.168.0.122 -j ACCEPT
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT
-A INPUT -m limit --limit 5/min -j LOG --log-prefix "iptables denied: " --log-level 7
- A Reject all other inbound - default deny unless explicitly allowed policy:
-A INPUT -j REJECT
-A FORWARD -j REJECT
COMMIT

These six rules below will block the vast majority of all sip scanner traffic that randomly scans the Internet. Use these rules in conjunction with Fail2Ban and you will be in good shape to avoid rogue attackers.

SIP scanner iptables block

iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "VaxSIPUserAgent" --algo bm
iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "friendly-scanner" --algo bm
iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "sipcli" --algo bm
iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "VaxSIPUserAgent" --algo bm
iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "friendly-scanner" --algo bm
iptables -I INPUT -j DROP -p udp --dport 5060 -m string --string "sipcli" --algo bm
Turn on the rules

`iptables-restore < ~/iptables.fs.rules`

The rules are now turned on. Test and make changes to your firewall as necessary.

Save the final rules

Once you are happy with your rules, save them.

`iptables-save > /etc/iptables.up.rules`

Load rules on boot

```bash
vim /etc/network/if-pre-up.d/iptables
#!/bin/bash
/sbin/iptables-restore < /etc/iptables.up.rules
chmod +x /etc/network/if-pre-up.d/iptables
```