

Importing info from Freeswitch HA

To provide high availability to protect from a single server meltdown, crash, or to have minimum downtime maintenance e.g. (1-4 seconds of lost audio, with no dropped calls) , use this configuration.

NOTE: This is not a cluster, each single machine must be able to handle the entire load itself.

Other than the system IP configuration file (with static internal / external IPs), and the heartbeat monitoring IP, the entire system should be identical, especially all of FreeSWITCH's files. That means you can upgrade FreeSWITCH on the backup, do a little testing, move over the traffic, and then rsync the entire FreeSWITCH directory to the other server.

Contents

- 1 Hardware requirements
- 2 Software requirements
- 3 Configuration
 - ◆ 3.1 HeartBeat/Pacemaker
 - ◆ 3.2 ODBC
 - ◆ 3.3 Sofia Profiles
 - ◆ 3.4 Voicemail
 - ◆ 3.5 XML_CDR Logs

Hardware requirements

- 2 servers (or VPS, preferably different physical machines)
- a 3rd, Floating IP

Software requirements

- IP Failover
 - ◆ Heartbeat to monitor the other server is responding
 - ◆ Pacemaker to monitor and move resources (floating IP, starting/restarting FreeSWITCH profiles, e.g. triggering "sofia recover"
 - ◇ FreeSWITCH monitoring OCF script
 - ◆ OR uCARP
- ODBC for the database for anything to be saved, with replication. For core, voicemail, and all profiles and mods. I'm using MySQL in master-master mode.

Configuration

HeartBeat/Pacemaker

/etc/init.d/freeswitch Add this to the list.. start, stop..

```
status)
    status_of_proc -p $PIDFILE freeswitch  && exit 0 || exit 3
    ;;
```

/etc/init.d/FSSofia - the "start" part might be a bad idea..

```
#!/bin/sh
#
# Description:   Starting / stopping FreeSWITCH Sofia SIP profiles
#               and send a 'sofia recover' after all profiles are started
#
# Author:       Leon de Rooij <leon@scarlet-internet.nl>
# License:      BSD
# Copyright:    (C) 2010 Leon de Rooij

#set -x

FS_CLI_PROG='/usr/local/freeswitch/bin/fs_cli'
FS_CLI_HOST='127.0.0.1'
FS_CLI_PORT='8021'
FS_CLI_PASS='ClueCon'
PROFILES='internal external'

usage() {
    echo "Usage: $0 profile1[,profile2[,etc]] {start|stop|status}"
    exit 1
}

fs_cli() {
    $FS_CLI_PROG -H $FS_CLI_HOST -P $FS_CLI_PORT -p $FS_CLI_PASS -x "$1"
}

sofia_profile_started() {
    fs_cli "sofia xmlstatus" | grep "<name>$1</name>" | wc -l
}

if [ $# != 1 ]; then
    usage
fi
#was 2 also.

#PROFILES=`echo $1 | tr ',' ' '`
CMD=$1
#was $2

case "$CMD" in
    'start')
        `etc/init.d/freeswitch status`
        if [ $? -eq 3 ]; then
            `/usr/local/freeswitch/bin/freeswitch -nonat`
```

Enterprise_deployment_IP_Failover

```
fi
# /etc/init.d/freeswitch start
for p in $PROFILES; do
    fs_cli "sofia profile $p start"
done
fs_cli "sofia recover"
;;
'stop')
# for p in $PROFILES; do
#     fs_cli "sofia profile $p stop"
# done
;;
'status')
for p in $PROFILES; do
    if [ `sofia_profile_started "$p" -eq 0 `]; then
        echo "$p DOWN"
        exit 3
    fi
done
echo "OK"
;;
*)
usage
;;
esac
```

My current "crm configure edit"

```
node $id="astarstt-e939-srat-870c-rastartsarsr" server1 \
    attributes standby="off"
node $id="astarstt-7203-arst-b5a7-esintsireant" server2 \
    attributes standby="off"
primitive fs lsb:FSSofia \
    op monitor interval="2" timeout="8" \
    meta target-role="Started"
primitive ip_shared ocf:heartbeat:IPaddr2 \
    params ip="$floating_ip" nic="eth0" \
    meta target-role="Started"
group HAServices ip_shared fs \
    meta target-role="Started"
property $id="cib-bootstrap-options" \
    dc-version="1.0.8-042548a451fce8400660f6031f4da6f0223dd5dd" \
    cluster-infrastructure="Heartbeat" \
    expected-quorum-votes="1" \
    stonith-enabled="false" \
    no-quorum-policy="ignore" \
    last-lrm-refresh="1299964019"
rsc_defaults $id="rsc-options" \
    resource-stickiness="100"
```

ODBC

- Please set up ODBC in the core
- Also, set up the DSN in limit.conf.xml if you plan on using limits

Sofia Profiles

In each profile that you want to be able to failover, you need to tell FreeSWITCH to track all channel states in the database...

```
<param name="track-calls" value="true"/>
```

And tell it which database to use that will be replicated.

```
<param name="odbc-dsn" value="dsn:username:password"/>
```

Also, you'll have to specify the SIP, RTP bind IP etc to be the floating IP, rather than the machine's auto-detected IP.

```
rtp-ip, sip-ip, presence-hosts, ext-rtp-ip, ext-sip-ip
```

You can leave the profile running binded to the floating IP that's not yet there. However, your OS may not allow it until you do:

```
On ubuntu: sysctl net.ipv4.ip_nonlocal_bind=1
Others:
sysctl to set net.ipv4.ip_nonlocal_bind = 1
or
echo 1 > /proc/sys/net/ipv4/ip_nonlocal_bind
```

Voicemail

Specify the ODBC-DSN to store the voicemail preferences and messages:

```
in: voicemail.conf.xml
<param name="odbc-dsn" value="dsn:username:password"/>
```

And, make sure that either

- Both servers have the same domain name in their configuration (will be if you rsync them, but the default may be the machine's IP)
- OR change the default voicemail storage location.

NOTE: There needs to be a mechanism to sync the voicemail greetings and messages. An rsync --update from both sides will keep it current, but will keep deleted voicemail files. A one-way rsync from the master with --delete is better, perhaps triggered with a hangup hook from the dialplan after transferring to voicemail, OR some sort of cron set up in pacemaker. Remember, this needs to be asynchronous-able, so that you can push updates to a server that was down and comes back up.

XML_CDR Logs

If you just write the log, that can be easily rsync'd without worry of deletions.

Enterprise_deployment_IP_Failover

If you use FusionPBX, please make sure you are using the web submission, so that the CDRs get written once directly into the archive folder, rather than getting written to the archive folder and THEN moved. That would make things confusing because we would need to keep track of the master to keep the xml logs in sync.