

# SIP Trunking using the Optimum Business SIP Trunk Adaptor and the Asterisk IP-PBX 13.2

## Table of Contents

1. Overview .....	3
2. Prerequisites .....	3
3. Asterisk PBX Configuration .....	3
3.1 Asterisk Configuration Files .....	4
3.2 SIP Trunk Configuration to the Optimum Business SIP Trunk Adaptor (Edgewater Networks 4550 series) .....	4
3.3 Inbound Trunk section .....	5
3.4 Outbound Trunk Section .....	6
3.5 SIP Phone/Extension Configuration .....	7
3.6 Dial plans, Auto-Attendants, and Parking lots .....	11
3.7 Parking Lot Configuration / features.conf .....	18
3.8 Asterisk Console Logging / Troubleshooting .....	18

## 1. Overview

The purpose of this configuration guide is to describe the steps needed to configure the Asterisk PBX for proper operation with Optimum Business SIP Trunking.

## 2. Prerequisites

Follow the instructions in the Optimum Business SIP Trunk Set-Up Guide left by the Optimum Business technician at installation. If you do not have the Set-Up Guide you can download it at [www.optimumbusiness.com/SIP](http://www.optimumbusiness.com/SIP).

This guide provides the configuration steps for both PBX registration and static or non-registration modes of PBX operation.

**Table 1 – PBX Information**

<b>Manufacturer:</b>	Open Source Asterisk
<b>Model:</b>	SIP
<b>Software Version:</b>	13.2
<b>Does the PBX send SIP Registration messages (Yes/No)?</b>	Yes
<b>Vendor Contact:</b>	irc.freenode.net #asterisk

## 3. Asterisk PBX Configuration

The steps below describe the basic configuration required to enable the Asterisk PBX to use Optimum Business SIP Trunking for inbound and outbound calling. Please refer to the Asterisk documentation for other advanced PBX features.

This configuration is based on Asterisk software version 13.2.

Many other PBX systems are based off of the Asterisk core, such as Trixbox, FreePBX, and AsteriskNOW as well as many others. Even though they may have GUI's, the core configuration files and syntax is the same, and the core functions and operations are the same. Just take into account that like most other UNIX/Linux programs, the core configuration file directories can be placed anywhere in the system.

This guide will go through the steps of installing Asterisk 13.2 on a Linux/GNU OS. Explain the configuration files and their specific purposes. Configuration of 3 phones on the LAN side, configuration of the inbound and outbound trunks to the Optimum Business SIP Trunk Adaptor, Dial Plans, Auto-Attendants, and Parking Lots, as well as basic console troubleshooting for the Asterisk system.

## 3.1 Asterisk Configuration Files

As part of this configuration guide there will be 3 conf files that will be explained and configured. The extensions.conf file, the sip.conf file, and the features.conf file.

### Extensions.conf

This conf file contains the extensions for phones and DID's the come into the PBX, dial plans, Auto-attendants, and call forwarding configurations.

### Sip.conf

This conf file contains the global register configuration to the SIP trunks, the inbound and outbound call settings, and the phone/extension configuration and registration settings.

### Features.conf

This is where the parking lot is configured.

## 3.2 SIP Trunk Configuration to the Optimum Business SIP Trunk Adaptor (Edgewater Networks 4550 series)

Within the sip.conf file resides the configuration for working with the SIP Trunk. All configurations in this file must go under the [General] section.

Add the register string, this is only required if the Asterisk PBX needs to register to the Optimum Business SIP Trunk Adaptor. The register string MUST come before any phone/extension or trunk configuration, and directly after the [General] section.

Format:

```
register => user:secret:@host[:port]/extension
```

Example register string:

```
register => 4085555555:p@ssw0rd@192.168.1.1:5060/4085555555
```

The DID listed here, 4085555555 is the pilot DID of the SIP Trunk Group, it is the Authentication Username that the Optimum Business SIP Trunk Adaptor looks for when a registration originates from the PBX. This can be changed to anything as long as the Optimum Business SIP Trunk Adaptor is changed to reflect these setting. Following it is a ":" to signify the next part of the registration parameters.

The next part is the Authentication Password the Optimum Business SIP Trunk Adaptor looks for when the PBX registers to the Optimum Business SIP Trunk Adaptor. Next is the IP/domain of the SIP server, the LAN side IP address of the Optimum Business SIP Trunk Adaptor acts as the SIP server to the PBX,

@192.168.1.1. The extension/DID that is attached to the end of the string is to indicate what user uri to use, by default it will be “s” if this is not defined.

This string will cause a register attempt to the Optimum Business SIP Trunk Adaptor with the Authentication Username of 4085555555 and password of p@ssw0rd to the SIP server address of 192.168.1.1. It attempts to register every 60 seconds by default.

## 3.3 Inbound Trunk section.

Name the inbound trunk section:

```
[ Inbound ]
```

State what type of connection this is. A SIP trunk is a “peer” in this instance:

```
type=peer
```

State the DTMF method to be used:

```
dtmfmode=inband
```

There are 4 options for DTMF; inband, RFC2833, SIP INFO, and auto, which will use whatever method is negotiated from the SIP Provider.

```
dtmfmode=auto
```

```
dtmfmode=inband
```

```
dtmfmode=rfc2833
```

```
dtmfmode=info
```

The inband option requires that ONLY the G.711 codec be used for audio streams (explained further down).

Define which codecs to be used with the audio streams, first disallow ALL codecs:

```
disallow=all
```

Then follow with the allowed codecs:

```
allow=ulaw
```

```
allow=alaw
```

Ul原因 and alaw is G.711.

Select a context name for your Inbound Trunk to reference to, this is case sensitive and is referred to in other conf files:

```
context=inbound
```

A very important option is to tell Asterisk if it is behind a NAT or if it is not behind a NAT. Even though the Optimum Business SIP Trunk Adaptor is NAT'ing the IP headers to and from Asterisk, the VoIP ALG built into the Optimum Business SIP Trunk Adaptor will deal with the proper header manipulations for SIP. Turn off NAT

in the Asterisk to prevent header manipulation conflicts:

```
nat=no
```

Specify the IP from which the SIP signaling will come from, this is the same as the SIP Server, set it to the Optimum Business SIP Trunk Adaptor's LAN IP address:

```
host=192.168.1.1
```

Set the port for Asterisk to bind/listen to:

```
bindport=5060
```

## 3.4 Outbound Trunk Section

Set the section name:

```
[ Outbound ]
```

Specify if the Outbound Trunk/peer should send SIP OPTION messages (keep alive) to the SIP Server (Optimum Business SIP Trunk Adaptor) to check and see if the SIP server is there and active, this is only useful if multiple outbound trunks are used, turn this off:

```
qualify=no
```

Set the connection type:

```
type=peer
```

Set the DTMF method to be used on outbound tones:

```
dtmfmode=inband
```

Set the audio codecs to be used and negotiated on outbound calls:

```
disallow=all
```

```
allow=ulaw
```

```
allow=alaw
```

Set the context, remember this is case sensitive and will be referenced later on:

```
context=outbound
```

Specify the Caller ID, the Caller ID Number, and the FROM user which will show up in the FROM field of all SIP messages originating from the Asterisk. The callerid can be set to anything within 16 characters, this is usually the name that shows up:

```
callerid=Anonymous
```

or

```
callerid=4085555555
```

Set the fromuser for the FROM header:

```
fromuser=4085555555
```

And the Caller ID Number:

```
cid_number=4085555555
```

Note: When an extension has a DID specified as the callerID/fromuser/cid\_number, these last three options will need to be removed to prevent overriding the DID assigned to the extension.

## 3.5 SIP Phone/Extension Configuration

The phone/extension configuration resides in the sip.conf file, directly after the trunk/peers configuration.

Name the phone/extension. Typically this is the extension assigned to the phone but it can also be a name:

```
[100]
```

Specify the type of connection, a phone/extension or device on the LAN side of the PBX is called a “friend” in this instance:

```
type=friend
```

Specify the IP address the phone/extension is at, most of the time phones get IP addresses from a DHCP server, if that’s the case, set the host option to dynamic:

```
host=dynamic
```

Set the context of the phone/extension, just like the trunks/peers this is case sensitive and referenced to later on:

```
context=phones
```

Set the registration username the phone/extension will use to register to the PBX:

```
username=100
```

Set the password the phone/extension will use to register to the PBX (note, it is CRITICAL to set a strong password to warn off hacking attempts):

```
secret=p@ssw0rd
```

Set the DTMF method used by the phone:

```
dtmfmode=inband
```

Set the NAT options for the phone/extension:

```
nat=no
```

Set the audio stream codecs for the phone/extension to negotiate:

```
disallow=all
```

```
allow=ulaw
```

```
allow=alaw
```

Set the allowed IP range where the phone/extension is permitted to come from, this is a good security step to prevent fraud:

```
deny=0.0.0.0/0  
permit=192.168.1.0/255.255.255.0
```

If this particular phone/extension will be attached to a DID, set the caller ID, caller ID number, and the FROM name to be used (if these options are set in the Outbound trunk section, the Outbound trunk settings will override the phone caller ID settings):

```
cid_number=4085555558  
callerid=4085555558  
fromuser=4085555558
```

Below is how it should look in the sip.conf file:

```
[100]  
type=friend  
host=dynamic  
context=phones  
username=100  
secret=p@ssw0rd  
dtmfmode=auto  
nat=no  
disallow=all  
allow=ulaw  
allow=alaw  
deny=0.0.0.0/0  
permit=192.168.1.0/255.255.255.0  
cid_number=4085555558  
callerid=4085555558  
fromuser=4085555558
```

Repeat for other phones/extensions.

This completes the configuration of the sip.conf file. This is how it should look:

```
[General]  
register => 4085555555:p@ssw0rd@192.168.1.1:5060  
  
[Inbound]  
type=peer  
dtmfmode=auto  
disallow=all
```



```
allow=ulaw
allow=alaw
insecure=invite,port
context=inbound
nat=no
host=192.168.1.1
bindport=5060

[Outbound]
qualify=no
type=peer
dtmfmode=inband
disallow=all
allow=ulaw
allow=alaw
insecure=invite,port
context=outbound
nat=no
host=192.168.1.1
cid_number=4085555555
callerid=Anonymous
fromuser=4085555555

[100]
type=friend
host=dynamic
context=phones
username=100
secret=p@ssw0rd
dtmfmode=auto
nat=no
disallow=all
allow=ulaw
allow=alaw
deny=0.0.0.0/0
permit=192.168.1.0/255.255.255.0
cid_number=4085555558
callerid=4085555558
fromuser=4085555558
```

```
[Bob]
type=friend
host=dynamic
context=phones
username=101
secret=p@ssw0rd
dtmfmode=auto
nat=no
disallow=all
allow=ulaw
allow=alaw
deny=0.0.0.0/0
permit=192.168.1.0/255.255.255.0
cid_number=4085555557
callerid=4085555557
fromuser=4085555557
```

```
[Fax]
type=friend
host=dynamic
context=phones
username=102
secret=p@ssw0rd
dtmfmode=auto
nat=no
disallow=all
allow=ulaw
allow=alaw
deny=0.0.0.0/0
permit=192.168.1.0/255.255.255.0
cid_number=4085555556
callerid=4085555556
fromuser=4085555556
```

## 3.6 Dial plans, Auto-Attendants, and Parking lots

Like the sip.conf file, extensions.conf needs to start with a [General] section, everything else will be configured after this section.

```
[General]
static=yes
writeprotect=no
clearglobalvars=no
```

To define the DID's that will come into the Asterisk from the SIP Server, use the context defined in the sip.conf file under the Inbound Trunk/peer section:

```
[inbound]
```

Tell asterisk what to do with the DID's when they come from the SIP Trunk:

```
exten => _4085555555,1,Answer
exten => _4085555555,n,GoTo(aa,s,1)
exten => _4085555555,n,Hangup( )
```

The first line, when a call comes into the Asterisk from the Inbound Trunk, and it is destined to 4085555555, answer it automatically. Second line, for the same number, send it to aa,s,1. "aa" is the Auto-Attendant that is setup further down. Third line simply says Hangup the call once the Auto-Attendant is done processing the call.

Notice the 1, then below that the n and then below that another n. This is the numbering scheme for the call flow. A 2, then a 3, could also be used.

Set up another DID, this time sending it to a phone/extension configured in the sip.conf file.

```
exten => _4085555558,1,Answer
exten => _4085555558,2,GoToIf( ${DB(CFIM2/100)} = 1)?CFA:NA)
exten => _4085555558,3(CFA),Set(CFA=${DB(CFIM/100)})
exten => _4085555558,4,Dial(SIP/Outbound/${CFA})
exten => _4085555558,5(NA),Dial(SIP/100,20)
exten => _4085555558,6,GoToIf( ${DB(CFBS2/100)} = 1)?CFB:NA2)
exten => _4085555558,7(CFB),Set(CFB=${DB(CFBS/100)})
exten => _4085555558,8,Dial(SIP/Outbound/${CFB})
exten => _4085555558,9(NA2),Busy
```

Same flow as before, except this time Dial SIP extension 100 and include the rules for Call Forwarding Always and Call Forwarding Busy.

Repeat for all DID's that the Asterisk will use.

Configure each phones extension, this is required so that Asterisk knows how each phone/extension can talk to each other, or where the phones/extensions are at in order to forward calls to them, make sure this section is named the same as the context section that the phones/extensions are under:

```
[ phones ]
```

```
exten => 100,1,Dial(SIP/100)
```

```
exten => 100,n,Hangup( )
```

```
exten => 101,1,Dial(SIP/Bob)
```

```
exten => 101,n,Hangup( )
```

```
exten => 102,1,Dial(SIP/Fax)
```

```
exten => 102,n,Hangup( )
```

An extension may have been named Bob or Fax, but the 3 digit extension is dialed, not the name of the extension. However the name is what's referenced when Asterisk looks for it.

Also add a line under this section for the parking lot, this will be explained later:

```
include => parkedcalls
```

If a phone/extension makes an outbound call, Asterisk needs to know which trunk to send it the call out through, it also needs to know which dialed digits are allowed.

Look at a single digit, if "0" was to be dialed to get an operator and sent to the Outbound Trunk/peer:

```
exten => _X,1,Answer( )
```

```
exten => _X,n,Log(NOTICE, Dialing out from ${CALLERID(all)}  
to ${EXTEN} through Outbound)
```

```
exten => _X,n,Dial(SIP/Outbound/${EXTEN} )
```

```
exten => _X,n,Playtones( congestion)
```

```
exten => _X,n,Hangup( )
```

The first line tells Asterisk to grab the call, it "Answers" the call. The second line says log the call to the console, this will be shown in the Asterisk console for debugging. The Third line says send the call to the Outbound trunk/peer section specified in the sip.conf file, this is NOT the outbound context, but rather the section name. The fourth line is play a "Not Available" tone if the call was not sent to the Outbound Trunk/peer, or if the trunk/peer rejected the call. The final line Hangs up the call.

Repeat for all dial plans in the following order, Single digit( 0 for the operator),

2 digits( 00 for the operator), 3 digit(911, 311, 411), Call Forwarding(\* and \*\*), local calls(xxx-xxxx, 555-1111), long distance(xxx-xxx-xxxx, 677-555-1111), long distance with 1(1-677-555-1111), and international (011).

Setup the Auto-Attendant.

```
[aa]

exten => s,1,Answer
exten => s,n,Background(/var/lib/asterisk/sounds/en/
PleaseDialExt)
exten => s,n,WaitExten
exten => 100,1,Dial(SIP/100)
exten => 100,n,Hangup
exten => 101,1,Dial(SIP/Bob)
exten => 101,n,Hangup( )
```

Line 1 picks up calls sent to the aa context.

Line 2 plays the sound file PleaseDialExt, which is located in /var/lib/asterisk/sounds/en, while it is being played, the Auto-Attendant is waiting for input from the caller.

Line 3 tells the auto-attendant to wait until there is a response from the caller.

Line 4 is if the user hits 100, forward the caller to extension 100

Line 6 is the same as line 4, but for extension 101(Bob).

Set up call forwarding.

```
; Unconditional Call Forward
exten => *_21*X.,1,Set(DB(CFIM/${CALLERID(num)}))=${REPLACE(EXTEN:4,*#-)}
exten => *_21*X.,2,Set(DB(CFIM2/${CALLERID(num)})=1)

exten => *_21*X.,n,Playback(/var/lib/asterisk/sounds/en/
added)
; audio file may vary, select one that exist
exten => *_21*X.,n,Hangup
exten => **21*,1,Set(DB(CFIM2/${CALLERID(num)})=0)
exten => **21*,n,Playback(/var/lib/asterisk/sounds/en/de-
activated)
; audio file may vary, select one that exist
exten = **21*,n,Hangup
```

```
; Call Forward on Busy or Unavailable
exten => *_61*X.,1,DBput(CFBS/${CALLERIDNUM})=${EXTEN:4})
exten => *_61*X.,2,Set(DB(CFBS2/${CALLERID(num)})=1)
exten => *_61*X.,n,Playback(/var/lib/asterisk/sounds/en/
added)
; audio file may vary, select one that exist
exten => *_61*X.,2,Hangup
exten => **61*X.,2,Set(DB(CFBS2/${CALLERID(num)})=0)
exten => **61*,n,Playback(/var/lib/asterisk/sounds/en/de-
activated)
; audio file may vary, select one that exist
exten => **61*,2,Hangup
```

The first 4 lines define Call Forward Always, this is setup when a phone/extension dials \*21\* then the number to forward the caller to. \*\*21\* removes the forward always.

The next 4 lines defines when callers should be forwarded only when the phone/extension being called is busy, this is setup when a phone/extension dials \*61\* then the number to forward the caller to. \*\*61\* removes the forward busy option.

This completes the extensions.conf config file. Below is the completed config file:

```
[General]
static=yes
writeprotect=no
clearglobalvars=no

[inbound]
exten => _4085555555,1,Answer
exten => _4085555555,n,GoTo(aa,s,1)
exten => _4085555555,n,Hangup( )

exten => _4085555558,1,Answer
exten => _4085555558,2,GoToIf($[${DB(CFIM2/${EXTEN})}] =
1]?CFA:NA)
exten => _4085555558,3(CFA),Set(CFA=${DB(CFIM/${EXTEN})})

exten => _4085555558,4,Dial(SIP/Outbound/${CFA})
exten => _4085555558,5(NA),Dial(SIP/100,20)
exten => _4085555558,6,GoToIf($[${DB(CFBS2/${EXTEN})}] =
1]?CFB:NA2)
```

```
exten => _4085555558,7(CFB),Set(CFB=${DB(CFBS/${EXTEN})})
exten => _4085555558,8,Dial(SIP/Outbound/${CFB})
exten => _4085555558,9(NA2),Busy

exten => _4085555557,1,Answer
exten => _4085555557,2,GoToIf($[${DB(CFIM2/${EXTEN})}]=1]?CFA:NA)
exten => _4085555557,3(CFA),Set(CFA=${DB(CFIM/${EXTEN})})
exten => _4085555557,4,Dial(SIP/Outbound/${CFA})
exten => _4085555557,5(NA),Dial(SIP/101,20)
exten => _4085555557,6,GoToIf($[${DB(CFBS2/${EXTEN})}]=1]?CFB:NA2)
exten => _4085555557,7(CFB),Set(CFB=${DB(CFBS/${EXTEN})})
exten => _4085555557,8,Dial(SIP/Outbound/${CFB})
exten => _4085555557,9(NA2),Busy

exten => _4085555556,1,Answer
exten => _4085555556,n,Dial(SIP/Fax)
exten => _4085555556,n,Hangup()

[phones]

exten => 100,1,Dial(SIP/100)
exten => 100,n,Hangup()

exten => 101,1,Dial(SIP/Bob)
exten => 101,n,Hangup()

exten => 102,1,Dial(SIP/Fax)
exten => 102,n,Hangup()

include => parkedcalls

exten => _X,1,Answer()
exten => _X,n,Log(NOTICE, Dialing out from ${CALLERID(all)}
to ${EXTEN} through Outbound)
exten => _X,n,Dial(SIP/Outbound/${EXTEN})
exten => _X,n,Playtones(congestion)
exten => _X,n,Hangup()

exten => _ZX,1,Answer()
exten => _ZX,n,Log(NOTICE, Dialing out from ${CALLERID(all)})
```

```
to ${EXTEN} through Outbound)

exten => _ZX,n,Dial(SIP/Outbound/${EXTEN})
exten => _ZX,n,Playtones(congestion)
exten => _ZX,n,Hangup()

; Unconditional Call Forward
exten => _*21*X.,1,Set(DB(CFIM/${CALLERID(num)})=${REPLACE(EXTEN:4,*#-)})
exten => _*21*X.,2,Set(DB(CFIM2/${CALLERID(num)})=1)
exten => _*21*X.,n,Playback(/var/lib/asterisk/sounds/en/added)
; audio file may vary, select one that exist
exten => _*21*X.,n,Hangup
exten => _**21*,1,Set(DB(CFIM2/${CALLERID(num)})=0)
exten => _**21*,n,Playback(/var/lib/asterisk/sounds/en/de-activated)
; audio file may vary, select one that exist
exten => _**21*,n,Hangup
; Call Forward on Busy or Unavailable
exten => _*61*X.,1,DBput(CFBS/${CALLERIDNUM}=${EXTEN:4})
exten => _*61*X.,2,Set(DB(CFBS2/${CALLERID(num)})=1)
exten => _*61*X.,n,Playback(/var/lib/asterisk/sounds/en/added)
; audio file may vary, select one that exist
exten => _*61*X.,2,Hangup
exten => _**61*X.,2,Set(DB(CFBS2/${CALLERID(num)})=0)
exten => _**61*,n,Playback(/var/lib/asterisk/sounds/en/de-activated)
; audio file may vary, select one that exist
exten => _**61*,2,Hangup

exten => _NXX,1,Answer()
exten => _NXX,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _NXX,n,Dial(SIP/Outbound/${EXTEN})
exten => _NXX,n,Playtones(congestion)
exten => _NXX,n,Hangup()
```



```
exten => _XXXX,1,Answer()
exten => _XXXX,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _XXXX,n,Dial(SIP/Outbound/${EXTEN})
exten => _XXXX,n,Playtones(congestion)
exten => _XXXX,n,Hangup()

exten => _NXXXXXX,1,Answer()
exten => _NXXXXXX,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _NXXXXXX,n,Dial(SIP/Outbound/${EXTEN})
exten => _NXXXXXX,n,Playtones(congestion)
exten => _NXXXXXX,n,Hangup()

exten => _NXXNXXXXXX,1,Answer()
exten => _NXXNXXXXXX,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _NXXNXXXXXX,n,Dial(SIP/Outbound/${EXTEN})
exten => _NXXNXXXXXX,n,Playtones(congestion)
exten => _NXXNXXXXXX,n,Hangup()

exten => _1NXXNXXXXXX,1,Answer()
exten => _1NXXNXXXXXX,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _1NXXNXXXXXX,n,Dial(SIP/Outbound/${EXTEN})
exten => _1NXXNXXXXXX,n,Playtones(congestion)
exten => _1NXXNXXXXXX,n,Hangup()

exten => _011.,1,Answer()
exten => _011.,n,Log(NOTICE, Dialing out from
${CALLERID(all)} to ${EXTEN} through Outbound)
exten => _011.,n,Dial(SIP/Outbound/${EXTEN})
exten => _011.,n,Playtones(congestion)
exten => _011.,n,Hangup()

[aa]

exten => s,1,Answer
exten => s,n,Background(/var/lib/asterisk/sounds/en/
PleaseDialExt)
```

```
exten => s,n,WaitExten
exten => 100,1,Dial(SIP/100)
exten => 100,n,Hangup
exten => 101,1,Dial(SIP/Bob)
exten => 101,n,Hangup()
```

## 3.7 Parking Lot Configuration / features.conf

The parking lot is configured under the features.conf section.

Like the extensions.conf and sip.conf files, features.conf needs to start with a [General] section and everything needs to be configured after this point.

```
[General]
```

Now put in the configuration for the parking lot:

```
parkext => 700 ; What extension to dial to park. Set per
parking lot.
parkpos => 701-720 ; What extensions to park calls on.
(default parking lot)
context => parkedcalls ; Which context parked calls are in
(default parking lot)
parkingtime => 15 ; Number of seconds a call can be parked
before returning.
comebacktoorigin = yes ; Setting this option configures the
behavior of call parking when the
courtesytone = beep ; Sound file to play to when someone
picks up a parked call
```

## 3.8 Asterisk Console Logging / Troubleshooting

To enable/enter the asterisk console:

```
# asterisk -rvvvvvvv
```

The number of v's indicates the verbosity of logging, 7 is the max.

The asterisk console can show the sip signaling that occurs between the phones and the SIP trunks, to enable this sip logging:

```
*CLI> sip set debug on
```

If changes were made to the conf files, reload the plans:

```
*CLI> reload
```