

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

Table of Contents

1. Overview 3

2. SIP Trunk Adaptor Set-up Instructions..... 3

3. Additional Set-up Information..... 7

4. International Calling..... 8

5. PBX Configuration 9

6. Asterisk Configuration Files 10

7. SIP Trunk Configuration 10

8. Inbound Trunk section 11

9. Outbound Trunk Section 12

10. SIP Phone/Extension Configuration 13

11. Dial plans, Auto-Attendants and Parking Lots..... 17

12. Parking Lot Configuration 24

13. Console Logging/Troubleshooting 24

1 Overview

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

2 SIP Trunk Adaptor Set-up Instructions

These instructions describe the steps needed to configure the LAN side of the Optimum Business SIP Trunk Adaptor.

Step 1:

Log on to the Optimum Business SIP Trunk Adaptor

1. Connect a PC to port 4 of the Optimum Business SIP Trunk Adaptor, the silver device labeled Edgewater Networks, 4550 series.



2. Open a Web browser and go to IP Address <http://10.10.200.1>. A login box will appear.
3. Enter login and password and click 'OK'.
Login: pbxinstall
Password: slptrunk



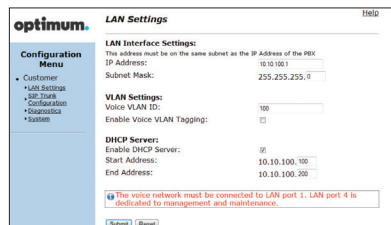
Step 2:

Click on the LAN Settings Link

1. Assign an IP Address to the LAN interface of the SIP Trunk Adaptor.
The IP address must be on the same subnet as the IP PBX. This changes the address on port 1 of the Optimum Business SIP Trunk Adaptor.
Note: This will become your local SIP proxy IP address. No other IP addresses will be provided by Cablevision.

2. Optional: Specify a VLAN for your voice traffic. Click the 'Enable Voice VLAN Tagging' check box. The default VLAN ID is 100.

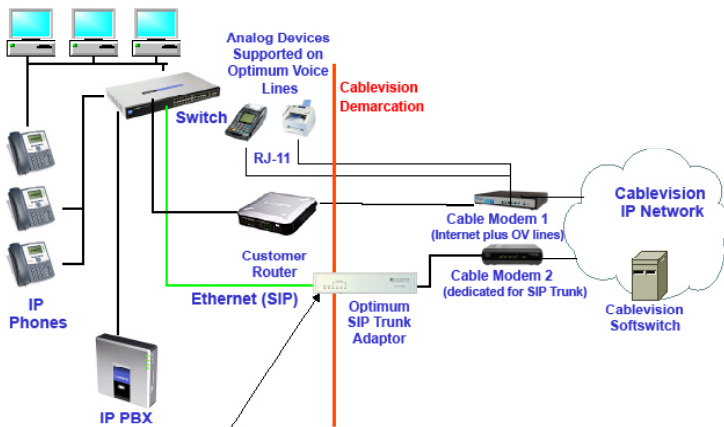
Note: VLAN 200 should not be used. It is dedicated to port 4 for management.



3. Optional: Enable the DHCP server. This will allow the SIP Trunk Adaptor to act as a DHCP server, which will provide IP addresses to the voice network, and create a dedicated voice LAN, as per diagram 2.

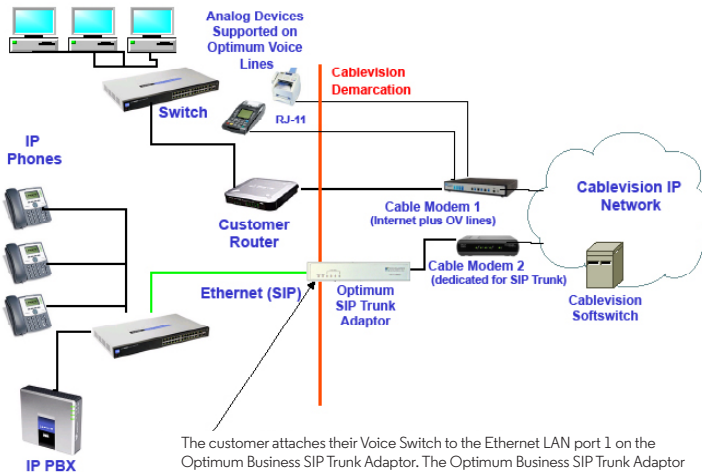
4. Click 'Submit'.

Diagram 1 SIP Trunk Adaptor for IP-PBXs
Example: Single LAN Configuration



Using a connection from the customer's LAN, the SIP Trunk Adaptor's address can be a statically assigned private IP address. It may not be assigned a Public IP address.

Diagram 2 SIP Trunk Adaptor for IP-PBXs
Example: Separate Voice and Data Networks Configuration



The customer attaches their Voice Switch to the Ethernet LAN port 1 on the Optimum Business SIP Trunk Adaptor. The Optimum Business SIP Trunk Adaptor can be enabled as a DHCP server to provide routing for the separate voice network.

Step 3:

Click on the SIP Trunk Configuration Link

1. Select your IP PBX make and model from the drop-down menu.
2. Specify how the IP PBX will register to the Optimum Business SIP Trunk Adaptor.
3. The Cablevision network only supports Inband DTMF. Click on the check box next to "Convert Inband DTMF" if you cannot configure your IP PBX to send out Inband DTMF. The DTMF tone duration generated by the phones and/or PBX may need to be increased from their default setting. Some phones and/or PBX have a default setting between 180ms to 200ms. This setting is too low. The recommended setting is 600ms.
4. Click 'Submit'.

Step 4:

Diagnostics Link

You can make a test call directly from your phone or use the test call application under the Diagnostics link.

Step 4 continued

Field	Description
Outbound Call Test TelephoneNumber	Specifies an outside phone number to which an outbound call will be initiated. The pilot telephone number of the SIP Trunk will be prepopulated.
Pilot Number	Displays the provisioned pilot number, which is used for outbound and inbound call tests.
Call	Initiates a call outbound to a telephone number entered or inbound to the pilot number displayed.
Inbound Call Test (radio button)	Indicates whether inbound test call will be enabled or disabled. If inbound test calls are enabled, calls made to the pilot number will be redirected to the test UA for fifteen minutes. When the pilot number is dialed, you will hear a test message play.
Submit	Enables or disables the inbound call test.
IP Address to Ping	Verifies basic connectivity to a networking device. Successful ping test results indicate that both physical and virtual path connections exist between the system and the test IP address.
Ping Button	Sends a ping to the IP address specified in the field "IP Address to Ping".
IP Address to Trace	Tracks the progress of a packet through the network. The packet can be tracked through the WAN or LAN interfaces of the adaptor.
Interface (radio button)	Indicates whether a packet will be tracked through the LAN or the WAN.
Traceroute Button	Initiates a traceroute to the specified IP address on either the LAN or the WAN.
Reset	Clears all fields and selections and allows you to enter new information. Reset applies to outbound call test, ping and traceroute.

3 Additional Set-up Information

Systems

optimum.

- Customer
 - LAN Settings
 - SIP Trunk Configuration
 - Diagnostics
 - System

System [Help](#)

Software Version:
Version 11.6.14.1 -- Fri Jan 4 17:49:28 PST 2013

Hostname:
5164939899

Model:
EdgeMarc 4552

Vendor:
Cablevision

LAN Interface MAC Address:
A8:70:A5:00:D8:18

Registration Status:
The ALG feature is registered. View [license key](#).

System Date:
02/29/2016 15:03:40 UTC

Change Password:

- [pbxinstall](#)

Field	Description
Pbxinstall Link	Select to change the default password for the pbxinstall login ID. Only the password can be changed. The login ID cannot be changed.

Password

optimum.

- Customer
 - LAN Settings
 - SIP Trunk Configuration
 - Diagnostics
 - System

Set Password [Help](#)

Change the GUI password by filling in the fields below. The password must be between 6 and 8 characters in length.

Username:

Current Password:

New Password:

Confirm Password:

Field	Description
Username	Specifies the username for which the password can be changed.
Current Password	Specifies the current password.
New Password	Specifies the new password.
Confirm Password	Confirms the new password.
Submit	Applies the settings configured on this page.
Reset	Clears all fields and selections and allows you to enter new information.

4 International Calling

Optimum Voice Business Trunking offers an optional International Calling Service for direct-dialed calls made from the Customer's business or from any phone via the Optimum Voice International Calling remote access number to destinations outside of the United States, Puerto Rico, Canada and the U.S. Virgin Islands at per minute rates. The Customer must login to the Optimum Business Account Center and activate the service on the Optimum Business Trunk Pilot telephone number to activate the service and manage the monthly International spending limit for the account.

Activating International calling on the Pilot TN will enable International calling for all Direct Inward Dial (DIDs) telephone numbers as well. Blocking International calling for one or more DIDs is managed by the customer directly from the PBX phone system configuration. To minimum the exposure to fraudulent calling, It is recommended to limit International calling capability to those DID's that require it and set up an account spending limit that reflects what is necessary to run the business.

It is the Customer (and/or the Customer Agent's) responsibility to properly secure the customer's PBX to prevent the PBX from being compromised and fraudulent calls from being made by unauthorized (internal or external) users. If fraudulent calls are detected, Cablevision reserves the right to disable International Calling until the PBX is properly secured by the customer.

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.

Table 1 – PBX Information

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

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.

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.

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
```

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.

1 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
```


11 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/deactivated)
; 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/deactivated)
; 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()
```

12 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
```

13 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
```