

Common configuration files

sip.conf	Configure trunks
users.conf	Configure users
extensions.conf	Dialplan configuration
iax.conf	
logger.conf	Files and log levels
confbridge.conf	Conference bridge configuration
modules.conf	
musiconhold.conf	Music on hold configuration
voicemail.conf	Voicemail configuration

Asterisk Console Commands

asterisk -r[vvvv]	Enter Asterisk console [verbose level]
sip show channels	Shows active channels
sip show registry	Display trunk status
sip show peers	Show peer status
core set verbose X	Changes the console verbosity level
core show translation	Displays transcoding latency between codecs
reload	Reread the configuration files, without restarting Asterisk
core restart now	Force Asterisk to restart immediately
core restart gracefully	Restarts Asterisk when calls are complete. New calls are denied
core restart when convenient	Restarts Asterisk when calls are complete. New calls are accepted

Pattern Matching

X	Digit from 0 to 9
Z	Digit from 1 to 9
N	Digit from 2 to 9
.	Wildcard - one or more characters
!	Wildcard - zero or more characters
[]	Numeric Range

Examples

_1XXX	match from 1000 to 1999
_12[0-5]X	match from 1200 to 1259
_12[015]	match 120, 121 and 125

Complex Match - All Numbers from 4683 to 5132

_468[3-9]	; 4683-4689
_469X	; 4690-4699
_4[7-9]XX	; 4700-4999
_50XX	; 5000-5099
_51[0-2]X	; 5100-5129
_513[0-2]	; 5130-5132

Dialplan Apps

Application	Description	Example
Answer()	Answer immediately	
Playback(-music)	Play music	
Dial(type/id-entifier, timeout)	Calls the type identifier number for timeout seconds	Dial(SIP/TrunkName/\${EXTEN:3}, 20)
VoiceMail(user@context)	Calls the user's voicemail user on context context	VoiceMail(\${EXTEN}@mycontext)
VoiceMailMain(user@context)	Allows you to check your voicemail	VoiceMailMain(\$CALLERID(num)@mycontext)
Goto(context, extension, priority)	Allows you to change context and/or number	Goto(otherContext, \${EXTEN:2})
Hangup()	Hang up	
NoOp(message)	Does nothing, useful for debugging	NoOp(Call from \${CALLERID(num)} to \${EXTEN})

Dialplan variables

<code>\${EXTEN}</code>	The number called
<code>\${CONTEXT}</code>	The name of the current context
<code>\${CALLERID(name)}</code>	The name of the caller
<code>\${CALLERID(num)}</code>	The caller's number

Dialplan Format

Contexts	Keeps different sections of the dialplan independent of each other	[users]
Extensions	The series of steps (each containing an application) defined for that extension	<code>exten => 100,1,Dial(SIP/alice)</code>
Priorities	An extension can have multiple steps (priorities), which are executed sequentially.	<code>exten => 100,1,Playback(tt-weasels)</code> <code>exten => 100,2,Voicemail(10)</code> <code>exten => 100,3,Hangup()</code>
n priority	Automatically increases a priority's value by 1,	<code>exten => 100,1,Playback(tt-weasels)</code> <code>exten => 100,n,Voicemail(10)</code> <code>exten => 100,n,Hangup()</code>
same => operator	Enables you to avoid writing the extension on each line, as long the extension remains the same.	<code>exten => 100,1,Playback(tt-weasels)</code> <code>same => n,Voicemail(10)</code> <code>same => n,Hangup()</code>
Applic- ations	the action that will be performed on the current channel.	<code>exten => 100,1,Dial(SIP/alice)</code>

Manipulating Variables

Variable format: `${variable_name[:offset[:length]]}` where

⚡ **offset**: how many digits to skip (optional)

⚡ **length**: number of digits to return (optional)

Examples: Assuming we've dialed 918005551234

1. Remove the first character of extension, save in "number" variable.

```
exten => _9X.,1,Set(number=${EXTEN:1})
```

2. Use a negative offset number to remove everything before the last four digits

```
exten => _9X.,1,Set(number=${EXTEN:-4})
```

3. Limit the number of characters from our offset position and only save 555

```
exten => _9X.,1,Set(number=${EXTEN:5:3})
```

4. The length value can also be used with a negative offset to save 555

```
exten => _9X.,1,Set(number=${EXTEN:-7:3})
```

5. Negative length value (-1) will remove # sign from the end of the string.

```
exten => _XXXX#,1,Set(pin=${EXTEN:0:-1})
```