

# Zapata.conf

## Synopsis

The `/etc/asterisk/zapata.conf` file contains parameters relating to channels provided by the zaptel interface layer. Channels must be defined in this file before they can be used by Asterisk. In addition, a number of features relating to Asterisk's operation of the channels may be configured here.

## Arrangement

The `/etc/asterisk/zapata.conf` file consists of keyword and value pairs. Keywords set parameters for the operation of channels. They may be boolean (yes/no) or contain values specific to the keyword. Most keywords set parameters for the operation of channels. Values remain in effect for all following channel definitions until they are overridden.

## Keywords

These keywords are available in `/etc/asterisk/zapata.conf`.

**context:** Defines the initial context for the channel. This will be the context available to the channel upon the initiation of a call. Note that contexts are an important part of maintaining site security. The initial context will govern the availability of extensions to a given channel. If an extension is placed in a different context from the initial context, that extension is unavailable to the caller.

*Note:* Careless use of contexts can allow open access to billable services and internal features.

**channel:** Define a channel or range of channels. Each channel definition will inherit all options stated ahead of it in the file.

## Examples

```
channel => 2
channel => 1-8
```

**group:** Allows a number of channels to be treated as one for the purpose of dialing. For dialing out, the channels will be called on a first available basis. For the purpose of ringing stations, all channels in the group will ring at once. Takes an integer.

**switchtype:** Sets the type of signalling used for a PRI line.

Values available for *switchtype* are:

|                  |               |
|------------------|---------------|
| <i>national:</i> | National ISDN |
| <i>dms100:</i>   | Nortel DMS100 |
| <i>4ess:</i>     | AT&T 4ESS     |
| <i>5ess:</i>     | Lucent 5ESS   |
| <i>euroisdn:</i> | EuroISDN      |

### Example

```
switchtype => national
```

**pri\_dialplan:** Sets an option required for some (rare) switches that require a dialplan parameter to be passed. This option is ignored by most PRI switches. It may be necessary on a few pieces of hardware. Valid options are unknown, local, private, national, and international. This option can almost always be left unset.

**signalling:** Sets the signaling type for following channel definitions. These should follow the channels as defined in /etc/zaptel.conf. Correct choices are based on the hardware available. Asterisk will fail to start if a channel signaling definition is incorrect or unworkable, if the statements do not match zaptel.conf, or if the device is not present or properly configured.

Values available for *signalling* are:

*Note:* Analog phone signalling can be a source of some confusion. FXS channels are signalled with FXO signalling, and vice versa. Asterisk 'talks' to internal devices as the opposite side. An FXO interface card is signalled with FXS signalling by Asterisk, and should be configured as such.

**fxo\_ks:** Use FXO Kewlstart signalling. Used to signal an FXS device within the system, which would normally drive a handset or other station device.

**fxs\_ks:** The opposite side of fxo\_ks. To signal an internal (or T1 connected) FXO device.

**fxo\_gs:** Use FXO groundstart signalling.

**fxs\_gs:** Use FXS groundstart signalling.

**fxo\_ls:** Use FXO loopstart signalling

**fxs\_ls:** Use FXS loopstart signalling

**pri\_cpe:** Use PRI signalling, customer equipment side. Used when terminating a PRI line into Asterisk channels.

**pri\_net:** Use PRI signalling, network side.

**em:** Use E&M signalling

**em\_w:** Use E&M wink signalling

**featd:** Feature Group D, Adtran compatible. For use with the Atlas and similar equipment made by Adtran.

**featdmf:** standard Feature Group D.

**featb:** Feature Group B

## Examples

```
signalling => fxs_ks  
signalling => featd
```

## Analog Call Progress

These items are used to attempt to emulate having a smarter line (like a PRI) that gives us call progress information, when using analog channels that don't pass us any digital information.

**busydetect:** Attempt to detect a standard busy signal on analog (FXS and FXO) or certain T1 signalling types (E&M, Wink, Feature Group D). This option can be used to determine when to hang up a call or to have Asterisk handle the busy condition internally. Takes ' yes' or ' no' .

**callprogress:** Used in combination with an incoming telephone line, enabling call progress will casue Asterisk to attempt to monitor the state of the call, and detect ringing, busy, and answered line. Note that this is not explicitly supported by the line technology, and is subject to errors, especially false answer detection. This only works with US phone tones at this time. Takes yes or no.

## Multi-link PPP Options (for PRI, requires network support)

These options are used to set adjust multi-link PPP options on PRI lines that support it. Multi-link PPP is a technology that allows channels on a PRI to be dynamically allocated between voice and data. Asterisk can take voice channels allocated to it, dial a Remote Access Server, and dump the channels into a special extension that delivers the channel to the zaptel data layer. See ZapRAS.

**minunused:** The minimum number of unused channels available. If there are fewer channels available, Asterisk will not attempt to bundle any channels and give them to the data connection. Takes an integer.

**minidle:** The minimum number of idle channels to bundle for the data link. Asterisk will keep this number of channels open for data, rather than taking them back for voice channels when needed. Takes an integer.

**idledial:** The number to dial as the idle number. This is typically the number to dial a Remote Access Server (RAS). Channels being idled for data will be sent to this extension. Takes an integer that does not conflict with any other extension in the dialplan, and has been defined as an idleext.

**idleext:** The extension to use as the idle extension. Takes a value in the form of ' exten@context' . Typically, the extension would be an extension to run the application ZapRAS.

## Example

```
minunused => 2  
minidle => 1  
idleext => 6999@idle  
idledial => 6999
```

## Timing Parameters

These keywords are used only with (non-PRI) T1 lines. All values are in milliseconds. These do not need to be set in most configurations, as the defaults work with most hardware. It has been noted that the common Adtran Atlas uses long winks of about 300 milliseconds, and channels from them should be configured accordingly.

***prewink:*** Sets the pre-wink timing.

***preflash:*** Sets the pre-flash timing.

***wink:*** Sets the wink timing.

***rxwink:*** Sets the receive wink timing.

***rxflash:*** Sets the receive flash timing.

***flash:*** Sets the flash timing.

***start:*** Sets the start timing.

***debounce:*** Sets the debounce timing.

### Examples

```
rxwink => 300
prewink => 20
```

## Caller ID Options

These keywords set various Caller ID options, including turning certain features off and setting the caller ID string for channels. Most caller ID features default to on.

The following three options are boolean (yes/no).

***usecallerid:*** Disables or enables caller ID transmission for the following channels.

***hidecallerid:*** Sets whether to hide outgoing caller ID. Defaults to no.

***calleridcallwaiting:*** Sets whether to receive caller ID during call waiting indication.

### Example

```
usecallerid => yes
hidecallerid => no
```

***callerid:*** Sets the caller ID string for a given channel. This keyword takes a properly formatted string containing the name and phone number to be supplied as caller ID. Caller can be set to ' asreceived' on trunk interfaces to pass the received callerid forward.

*Note:* Caller ID can only be transmitted to the public phone network with supported hardware, such as a PRI. It is not possible to set external caller ID on analog lines. On supported systems, the phone company only receives the number, and supplies the name from their records.

## Examples

```
callerid => "Mark Spencer" <256 428-6000>  
callerid => asreceived
```

## Other Call Feature Options

These options enable or disable the availability of advanced call features offered by Asterisk such as three-way calling and call forwarding on FXS interfaces. All of these options are boolean (yes/no).

***threewaycalling:*** Sets whether to allow three-way calling from the channel.

***cancallforward:*** Disables or enables call forwarding. Call forwarding is activated with \*72 and deactivated with \*73.

***transfer:*** Disables or enables call transferring.

***immediate:*** Sets whether asterisk should answer the channel immediately upon pickup, without waiting for input.

***ads:*** Sets the channel to be treated as an ADSI. The ADSI specification is system similar to caller ID to pass encoded information to an analog handset. It allows the creation of interactive visual menus on an LED display, offering access to services such as voicemail through a textual or (low-resolution) graphical interface.

## Audio Quality Tuning Options

These options adjust certain parameters of Asterisk that affect the audio quality of Zapata channels.

***echocancel:*** Disable or enable echo cancellation. In almost every configuration it is recommended that this be left on (or left unstated, as the default is always on.) Takes ' yes' or ' no' .

***echocancelwhenbridged:?*** Enables or disables echo cancellation during a bridged call. Should be set on or left unset. Takes ' yes' or ' no' .

***rxgain:*** Adjusts receive gain. This can be used to raise or lower the incoming volume to compensate for hardware differences. Takes a percentage of capacity, from -100% to +100%

***txgain:*** Adjusts transmit. This can be used to raise or lower the outgoing volume to compensate for hardware differences. Takes the same argument as rxgain.

## Examples

```
echocancel => yes  
echocancelwhenbridged => no  
rxgain => 20%
```

## Call Logging Options

These options change the way calls are recorded in the call detail records generated by Asterisk.

**amaflags:** Sets the AMA flags, affecting the categorization of entries in the call detail records. Accepts these values:

*billing:* Mark the entry for billing

*documentation:* Mark the entry for documentation.

*omit:* Do not record calls.

*default:* Sets the system default.

## Miscellaneous Options

There are a few other keywords that don't fit neatly into the previous categories.

**mailbox:** This keyword can be set to allow Asterisk to offer an audible (and visual, if supported by the handset) message waiting indication when the station handset is picked up. When the *mailbox* keyword is defined and an unheard message exists in the associated Inbox, Asterisk will produce a stutter dialtone for one seconds after the phone is picked up. On supported hardware, the message waiting light will be activated. Takes as an argument a mailbox number (which must be defined in voicemail.conf).

**language:** Turn on internationalization and set the language. This feature will set all system messages to a given language. Though the feature is prepared, English is the only language that has been completely recorded for the default Asterisk installation.

**stripmsd:** Strip the ' Most Significant Digit,' the first digit or digits from all calls originated on the channels. Takes as an argument the number of digits to strip. This option is deprecated, see the application ' StripMSD' for this functionality.

## Complete File Example

This is a complete example of a functional zapata.conf file. It is based on an 8 FXO by 16 FXS T1 channel bank.

```
[channels]

;set the FXO's in a group so we can dial out of them
;on a first-available basis

group => 1

;set the correct context for our dialout lines

context => pstn

;set the signalling (remember that we signal fxs channels
```

```
;with fxo, and vice versa)
signalling => fxs_ls
;set the AMA flags for clarity in the logs
amaflags => documentation
;define the channels that will be covered by the previous
;declarations (in this case all of our FXO's)
channel => 1-8
;reset the group, so we don't send outgoing calls to
;the internal lines
group => 2
;change the context, so we can allow greater access to
;services to internal users
context => internal
;set the signalling on the station lines (fxs)
signalling => fxo_ks
;set a mailbox number on the following channels
mailbox => 1234
;set the callerid string (though since we don't have a PRI
;it's only seen inside, not on the PSTN.)
callerid => "Dave Schools" <256 555 1234>
;and state the channel this will apply to
channel => 9
;continue and state more channels with mailbox indication
;and caller id strings
mailbox => 1235
callerid => "Michael Houser" <256 555 1235>
channel => 10
mailbox => 1236
callerid => "John Bell" <256 555 1236>
channel => 11
mailbox => 1237
callerid => "Grace Slick" <256 555 1237>
channel => 12
;remember the downward inheritance of options.
;if the next channel doesn't have a voicemail box, we need
```

```
;to set a garbage string, or he'll know whenever
;Grace has a message. Also the callerid should be
;nulled as well

mailbox => 0000
callerid => ""<>

;define a bunch of channels with no other options

channel => 13-22

;Put this phone in a different context, so we can give it
;a different initial dialplan...perhaps a lobby phone
;with public access

context => lobby
callerid => "Lobby" <5000>
channel => 23
;and turn the callerid off

callerid => ""<>

;we can create a 'hotline' phone by placing a phone in a special context
;and setting it to answer immediately. In extensions.conf we can route
;the phone to an IVR, direct to security, or make it call Steak-Out

context => hotline
immediate => yes
channel => 24
```