

# PSAM 2.4 Asterisk advanced configurations

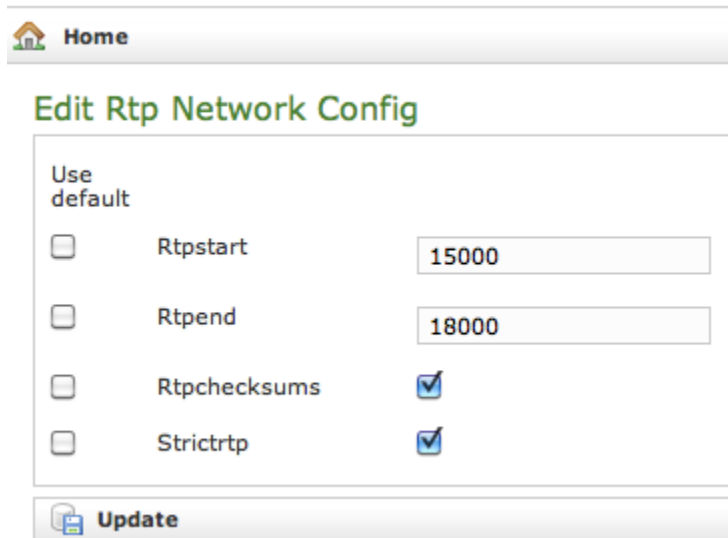
Here follow some advanced configuration settings about the PrivateServer behavior.

## 2.4.1 SIP/TLS

SIP/TLS is about configuring the encrypted communication channel among PrivateServer and its clients. The configuration form is reachable by the **SIP/TLS** main menu entry. From the editing form you can set up the cypher list of the PrivateServer. This is the list of accepted cipher suite, using OpenSSL format. Check at [http://www.openssl.org/docs/apps/ciphers.html#CIPHER\\_LIST\\_FORMAT](http://www.openssl.org/docs/apps/ciphers.html#CIPHER_LIST_FORMAT). Usually you can leave the default values.

## 2.4.2 RTP

"The **Real-time Transport Protocol (RTP)** defines a standardized packet format for delivering audio and video over IP networks" (quote from Wikipedia).



Use default		
<input type="checkbox"/>	Rtpstart	15000
<input type="checkbox"/>	Rtpend	18000
<input checked="" type="checkbox"/>	Rtpchecksums	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	Strict RTP	<input checked="" type="checkbox"/>


 **Update**

Figure 1. RTP configuration form

In this form that you get by the **RTP** main menu entry, you can set up the voice transport features. **Rtpstart** and **Rtpend** are the number of RTP ports available for the calls.



Each call uses 4 ports, thus you can do your math on the RTP number necessary in your configuration multiplying the number of foreseen concurrent calls for 4.

In the example shown in [Figure 1. RTP configuration form](#) you see:

$18000 - 15000 = 3000$  ports available. This means  $3000/4 = 750$  concurrent calls threshold.

The **Rtpchecksums** enable the application checksum over UDP encrypted voice transmission. This is an error detection commodity, which adds 16 bits per packets payload.

**Strict RTP** Enables the strict RTP protection. This will drop RTP packets that do not come from the source of the RTP stream. This option is disabled by default.

## 2.4.3 Jitter Buffer

**Jitter** is the undesired deviation from true periodicity of an assumed periodic voice streaming. The consequences of jitter, often called *jittering*, are a voice communication with holes in it or stirring metal voice effect. Mostly on a 3/4G network (and in general in a mobile network environment), the jitter is a sensible problem to face. To avoid jitter issues a **jitter buffer** is implemented in PrivateServer.

Home

## Edit Jitter Buffer Config

Use default

<input type="checkbox"/>	Enable	<input checked="" type="checkbox"/>
<input type="checkbox"/>	Force	<input checked="" type="checkbox"/>
<input type="checkbox"/>	MaxSize	500
<input type="checkbox"/>	Resync Threshold	5000
<input type="checkbox"/>	Impl	FIXED
<input type="checkbox"/>	Log	<input checked="" type="checkbox"/>

Update

Figure 2. Jitter Buffer configuration form

The **jitter buffer** is an elastic buffer in which the signal is temporarily stored and then retransmitted at a rate based on the average rate of the incoming signal. Checking the **Enable** checkbox you allow PrivateServer to perform such jitter buffering in general, when needed.

The **Force** option obliges PrivateServer to create the jitter buffer over any new communication channel.

**MaxSize** is the max length of the jitterbuffer in milliseconds.

**Resync Threshold** is the buffer synchronization threshold. It's useful to improve the quality of the voice, with big jumps in/broken timestamps. Defaults to 1000.

**Impl** is the Jitterbuffer implementation, used on the receiving side of a SIP channel. Two implementations are currently available:

- FIXED (with size always equals to jbmmaxsize)
- ADAPTIVE (with variable size)

Defaults to FIXED.

**Log** enables jitterbuffer frame logging. Defaults to "no".

Logged in as **admin** - [Logout](#)

**The following services need to be restarted to apply your changes: asterisk.**

Home

## Edit Jitter Buffer Config

**Jitter Buffer Config updated**

Figure 3. The Management console asks to restart the service

To apply your changes just press the **Update** button and the management interface will ask you to restart the asterisk service in order to apply your new configuration, as shown in

### 2.4.4 Obfuscation

The **Obfuscation** is an internal VoIP communication stealth mode. It is useful to avoid QoS (Quality of Service) checks on VoIP as it masks the data.



This practice is legal if you are not fooling your mobile provider or cheating your network administrator.



Home

## Edit Obfuscation Config

Mode

OFF



Key

0



Update



Update Accounts

The configuration is quite simple. **Mode** enables/disables the obfuscation mode.

**Key** is a shared numeric key to be reported on the clients configuration as well.



To avoid calls problems such as abruptly interrupted calls you make sure the obfuscation mode and key are equally set up on the server and the clients.