



Application Note

Micronet VoIP Solution with Asterisk

1. Introduction

This is the document for the applications between Micronet units and Asterisk IP PBX. It will show you some basic configurations in our units to work with Asterisk for some extra functions and also something you have to be note while you need these kinds of functions.

2. Before the configurations

Please get the detail info about the firmware version of our units and Asterisk:

Model Name	Asterisk
Firmware version	1.0.5
CPU	P4-2.5G
RAM	512M
OS	Red Hat 9.0

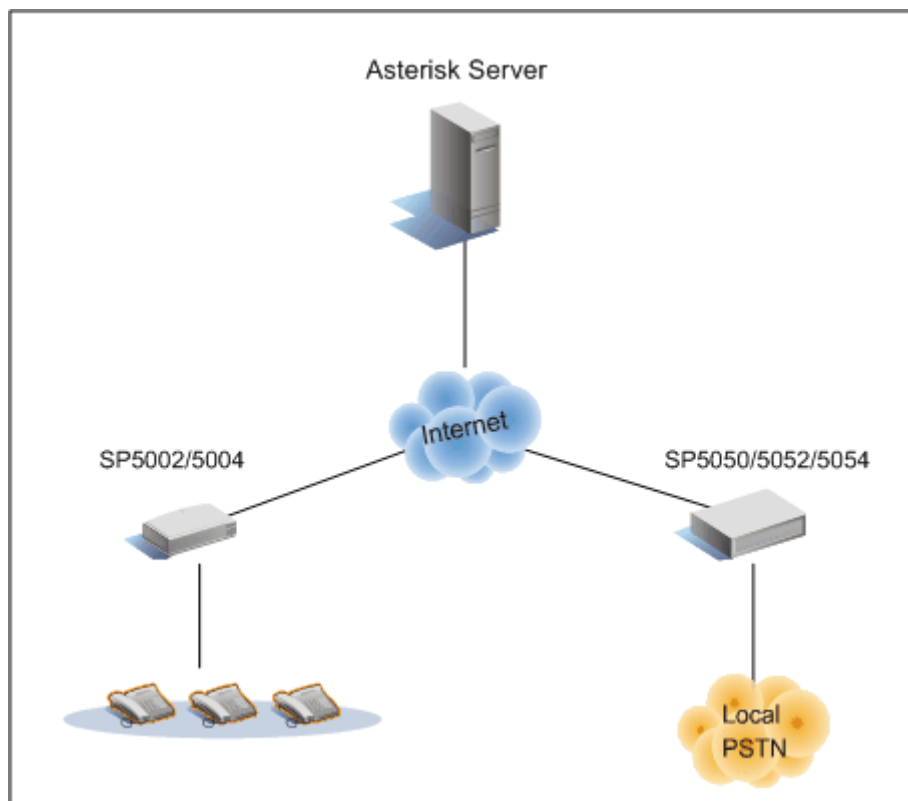
Model	SP5002/5004	SP5012/5014	SP5050/5052/5054
Version	v.108a	v.103	v.104

Before you start, please check out the hardware specification of Asterisk from the Asterisk's web site: <http://www.asterisk.org>

3. Registration with Asterisk

A file named “sip.conf” could manage the entire sip setting for the client or the server. User could define this for the registration with the client, including the phone number, account, password...etc. Please find out this file with the location /etc/asterisk and get the more detail info from the technical support web site of Asterisk.

There is nothing special for the configurations in Asterisk for the registration from Micronet VoIP units. The settings for Asterisk are as same as the other proxies.



3.1 Registration with the Authorization

The following is the example for the configurations in Asterisk with the registration from Micronet unit:

```
[886001]
type=friend
username=886001
secret=886001
host=dynamic
canreinvite=yes
context=Max
```

- 886001

Please also assign the phone number which you like in this configuration table.

- Type

According to the explanations from the documents of sip.conf file, there are three types for the client registrations. Friend is a device that can both receive and send calls through the Asterisk; Peer is a device which could receive calls from the Asterisk; User is a device that could make calls through Asterisk. This makes sense for most desk handsets and other devices. If unsure, please probably set the type to the friend value.

- username

This option sets the username that the Asterisk attempts to connect when a call is received. Used when for some reasons the value is not the same as the username the client registered.

- secret

Setting the password for the client. It takes an alphanumeric string.

- host

Setting the IP address or resolvable host name of the device. This can alternately be set to "dynamic" in which case the host is expected to come from any IP address.

- canreinvite

This option is used to tell the server to never issue a re-invite to the client.

- context

This is a key word for the dialing plan from asterisk. Please define this to match the "extensions.conf" file from Asterisk. For the sip info, it was managed by the

“sip.conf” file but it has to work with the “extensions.conf” file which is the dialing plan management.

According to the configurations in Asterisk, please check out the configurations as following:

```
usr/config$ sip -px 192.168.11.200 (The IP Address of Asterisk)
usr/config$ sip -line1 886001
usr/config$ security -line 1 -name 886001 -password 886001
usr/config$ commit
usr/config$ reboot
```

Note: Line number defined in the “sip.conf” file should be the same with account name.

3.2 Registration behind NAT

Users could define the NAT flag for the configurations with the client to resolve the application while Asterisk is outside the NAT and clients inside the NAT connecting with Asterisk. Please check out the example as the following:

```
[886001]
type=friend
username=886001
secret=886001
host=dynamic
canreinvite=yes
context=Max
nat=yes
```

For the NAT flag, it could be set with four types with this flag. Please get more detail info from the Asterisk wi-ki web site as following:

<http://www.voip-info.org/tiki-index.php?page=Asterisk+sip+nat>

Please pay more attentions on this function. All the NAT function was

controlled by the Asterisk not Micronet units. All the RTP packets for the voice will be passed through the Asterisk, so please check out the available codec for this application in the configurations of Asterisk.

According to the experiment in our side, we were using the codec G711 for this kind of application only. Please check out this from the Asterisk web site before you need this function.

4. Making the basic calls

For the configurations in the Registration chapter, it could make the registrations successful but the client couldn't make calls without another setting in Asterisk. As Asterisk is just like an IP PBX, users have to configure the "extensions.conf" file which could be found out from the location /etc/asterisk. All the dialing plans were controlled by this file and before the dialing from the client side, please configure this first.

The "extensions.conf" file lays out the dial-plan, bringing channels together with applications and service. This file features extension matching logic and intelligent call routing logic.

This file could support the some kind of protocol for the dialing, such like the Zap, IAX, and SIP...etc. In the setting rule, it could also support more parameters for the call making, such like the answering the calls automatically, hanging up automatically, playing IVR, and dialing the number...etc. It could also define some kind of parameters for this call, such as the color ring back tone, timer for timeout...etc. So please get more detail info from the Asterisk we site if you need more function with this extensions file.

4.1 Making calls to the registered clients

This has to be worked with the "sip.conf" while you try to configure the "extensions.conf" file for the dialing plan.

The following example was according to the sip configurations in the first chapter:

In "sip.conf" file,

```
[886001] ----- B
type=friend
username=886001
secret=886001
```

```
host=dynamic
canreinvite=yes
context=Max ----- A
nat=yes
```

The extensions.conf was based on the context name in the sip.conf file (B flag). Please check out the example in extensions.conf as following:

In “extensions.conf” file,

```
[Max]
exten => 886001,1,Dial(SIP/886001,20,Ttm)
```

While the clients send the number 886001 to Asterisk, it will find out the matching rule in the extensions.conf and decide the destination here. The number 886001 will be sent to the name 886001 (A flag) with the SIP protocol. The first emotion is dialing and timer for timeout is 20s. Regarding to the other flags just like “Ttm”, please get more detail info from the Asterisk web site about the parameters using in extensions.conf file. “T” is for the Absolute timeout, “t” is for the timeout and “m” is for the coloring ring back tone function.

Please get more or detail about this parameters using as following:

<http://www.voip-info.org/tiki-index.php?page=Asterisk+standard+extensions>

4.2 Making calls to the none-registered clients

Asterisk could also send the calls to the clients which were not registered on it. This could be used like set the calls to the Trunk gateway or FXO unit for the outbound calls. Please check out the example as following:

```
[Max]
exten => _0XXXXXXXX,1,Dial(SIP/${EXTEN}@61.219.198.203,20)
```

The calls will be sent to the client which was located with the ip address 61.219.198.203 with the prefix number is 0 and 10 digits. This syntax was

followed the extension document from the Asterisk web site. The parameter `#{EXTEN}` will capture the real numbers which was be sent from the client side.

For this configuration, please also checkout the dialing plan in the Trunk or FXO gateway. In FXO gateway (SP5050/5052/5054), because of the route table configuration, this call will be passed through the FXO port to the PSTN directly. This is not necessary for other special configurations in Micronet FXO units.

4.3 Auto IVR function in Asterisk

As the Asterisk is just like the IP PBX, it could also provide the IVR for the incoming calls. This chapter will introduce how to make it work.

For the IVR function between Asterisk and Micronet units, the calling party will get the IVR from Asterisk while they make calls from Micronet units. This function was provided by Asterisk and users have to enable this function in Asterisk before using it.

This function was also controlled by the file named “extensions.conf”. Please check out the following configurations in Asterisk about this function:

```
[mainmenu]
exten => 9,1,Answer
exten => 9,2,Background(thanks)
exten => 9,3,WaitExten
include => Max
```

While the client send the number 9 to Asterisk, the first emotion in Asterisk is answering this call, the second is playing the IVR which was named “thanks” and waiting for the extension number after the IVR.

In FXS gateway, users could dial the number “9” to get it. This is the normal dialing and users didn’t have any special configurations in Micronet unit.

In FXO gateway, it could pass through the incoming calls from the PSTN side to the IVR of Asterisk directly. It depends on the hotline function in FXO gateway.

Please check out the configurations as following in FXS gateway:

```
usr/config$ sysconf -service 1
usr/config$ bureau -hotline 1 61.33.33.33 (The IP address of Asterisk)
```

Please check out the configurations as following in FXO gateway:

```
usr/config$ line -config 1 hotline 9
```

After users dial the PSTN number into the FXO port, FXO gateway will enable the dial out the number "9" to the IP side automatically. Users could get the IVR from Asterisk while Asterisk answers this call.

Please pay more attentions for the third command in Asterisk. Asterisk will wait for the extension number from the client side, that's meaning it will detect the DTMF. For the DTMF signal detection, Asterisk could support the In-Band, RFC2833 and SIP info. Please make sure about this in the configuration file which was named sip.conf.

Note:

- There is known issue in IVR mode. Users will hear the click-click sound while connecting with FXO device with version 104. It is due to CNG function in FXO device. Please contact with sales updated version or get the new one.
- For the updated firmware version, please disable this function to avoid this situation:
sysconf -silence 0

Date: 2005/10/30