



## Rapport écrit TP Asterisk

### TP 2 – Configuration SIP et DialPlan.

#### 1. La CLI:

Pour afficher la version d'Asterisk : core show version

Pour afficher la liste des modules installés : feature list (ou module show)

#### 2. Configuration d'un poste SIP

1- Afin de permettre la connexion de poste (paramètres globaux) il faut configurer notre domaine (domain = 192.168.33.138) et le port d'écoute (bindport = 5060).

2- Pour configurer des comptes SIP sur Asterisk:

Dans le fichier /etc/asterisk/sip.conf

[moncontexte]

/\*déclaration d'un téléphone IP\*/

[200] // numéro de téléphone

type=friend // type de contact

secret=tplinux // mot de passe

qualify=yes // latence

nat=no // pas de NAT entre les téléphones et le serveur

host=dynamic // adresses IP récupéré par le protocole DHCP

canreinvite=no // tous les flux multimédias passent par le serveur asterisk

context=moncontexte // permet de définir des zones ayant des droits différents

mailbox=200@maboitevocale // numéro de la boite vocal associé

3- Pour recharger le module SIP afin de relire la configuration modifiée il suffit de taper la commande « sip reload »

Reloading SIP

== Parsing '/etc/asterisk/sip.conf': Found

== Parsing '/etc/asterisk/users.conf': Found

== Parsing '/etc/asterisk/sip\_notify.conf': Found

4- Domain, proxy, user, password, IP, ...

5- Asterisk sait le faire maintenant.

6- La commande qui permet d'afficher les informations des transactions SIP est sip set debug.

### 3. Configuration de base du plan de numérotation

1- Les contextes existants sont :

ael-default ael-demo ael-dundi-e164-canonical ael-dundi-e164-customers ael-dundi-e164-local ael-dundi-e164-lookup	dundi-e164-lookup dundi-e164-switch
ael-dundi-e164-switch ael-dundi-e164-via-pstn ael-iaxprovider	dundi-e164-via-pstn iaxprovider iaxtel700 international local longdistance macro-ael-dundi-e164 macro-ael-std-exten-ael macro-dundi-e164
ael-iaxtel700 ael-international ael-local	macro-page macro-stdexten macro-stdPrivacyexten
ael-longdistance ael-trunkint ael-trunkld ael-trunklocal ael-trunktollfree default	macro-trunkdial moncontexte page trunkint trunkld trunklocal trunktollfree parkedcalls
demo dundi-e164-canonical dundi-e164-customers	
dundi-e164-local	

2- Dans le fichier /etc/asterisk/extension.conf

**[moncontexte]** ; réglage appliqué à la zone « moncontexte »

**exten=> 200,1,dial(SIP/200,10)** ; lorsque que le composant le 200 en priorité (1) on // appelle via le protocole SIP le numéro 200 (SIP/200) au bout de dix seconde

**exten=> 200,2,voicemail (200@moncontexte)** ; permet de rediriger l'appel vers la messagerie

**exten=>200,3,hangup** ; si on ne peut pas joindre notre destinataire on raccroche (priorité=2)

**exten=>700,1,voicemailmain(\${CALLERID(num)})** ; permet de configurer le numéro de la boîte mail (le paramètre « \${CALLERID(num)} » évite de rentrer le login de l'utilisateur.

*Mais si on veut consulter sa boîte vocal sur une autre machine on élève se paramètre et la boîte vocale demandera alors de taper un login*

Dans le fichier /etc/asterisk/voicemail.conf

[Default]

200=> 0000,user 1,200@maboitevocal

Le premier paramètre (0000) est le mot de passe de la messagerie de l'utilisateur, le second (user 1) est le nom de l'utilisateur, le troisième sa boîte mail

Un mail sera envoyé à l'adresse mail (200@maboitevocal) lorsqu'un nouveau message sera laissé sur le répondeur.

3,4 et 5- Lorsque l'on ne se trouve pas dans le contexte « eleve » (**et qu'on n'est pas dans un contexte en commun**) on ne peut appeler un utilisateur présent dans ce dernier.

#### 4. Pattern Matching

```
exten=> 200,1,dial(SIP/200)
exten=> 200,2,voicemail (300@moncontexte)
exten=>200,3,hangup
```

et

```
exten=> 300,1,dial(SIP/300)
exten=> 300,2,voicemail (300@moncontexte)
exten=> 300,3,hangup
```

deviens

```
exten=> _XXX,1,dial(SIP/${EXTEN})
exten=> _XXX,2,voicemail (${EXTEN}@moncontexte)
exten=> _XXX,3,hangup
```

### TP 3 – Plan Asterisk et la messagerie d'entreprise.

#### 1. Boites Vocales

1 - On édite le fichier voicemail.conf afin de créer la boîte de messagerie de chaque poste.

```
[default]
100 => 100,100
101 => 101,101
```

2 - Afin d'accéder à la messagerie, on lui attribue l'extension 700 dans le fichier extensions.conf.

3. On laisse un message dans la messagerie d'un correspondant. On s'aperçoit que celui-ci est stocké dans le répertoire /var/spool/asterisk/voicemail/default/<correspondant> sous la forme d'un fichier .WAV sur le serveur Asterisk.

```
exten => 700,1,VoiceMailMain()
```

#### 2. Stockage IMAP et messagerie convergente

1 - Afin de permettre à Asterisk de stocker les messages sur un serveur IMAP, on installe tout d'abord un serveur Dovecot sur un des PC et on le configure ainsi.

```
protocols = imap imaps
log_timestamp = "%Y-%m-%d %H:%M:%S "
mail_privileged_group = mail
disable_plaintext_auth = no
protocol imap {
}
protocol managesieve {
  sieve=~/.dovecot.sieve
  sieve_storage=~/.sieve
```

```

}
auth default {
  mechanisms = plain
  user = root

  userdb passwd {
  }
  passdb passwd-file {
    args = /etc/passwd.dovecot
  }
}
dict {
}
plugin {
}

```

Le fichier /etc/passwd.dovecot contient les mots de passe en clair des utilisateurs :

```
user:{PLAIN}user
```

Enfin, on teste le serveur afin de voir si tout est correcte à l'aide d'une session telnet sur le port IMAP (143).

```

# telnet 192.168.20.1 143
Trying 192.168.20.1...
Connected to 192.168.20.1.
Escape character is '^]'.
* OK Dovecot ready.
a login user user
a OK Logged in.
c list "" *
* LIST (\NoInferiors \UnMarked) "/" "INBOX"
c OK List completed.

```

Le serveur répond aux requêtes, tout est correct.

2 - Il s'agit maintenant de configurer Asterix. Il faut tout d'abord charger le module de messagerie prenant en charge le protocole IMAP.

```

asterisk1*CLI> module load app_voicemail_imap.so
[Jun 11 22:40:42] WARNING[3230]: pbx.c:2981 ast_register_application: Already have an
application 'VoiceMail'
[Jun 11 22:40:42] WARNING[3230]: pbx.c:2981 ast_register_application: Already have an
application 'VoiceMailMain'
[Jun 11 22:40:42] WARNING[3230]: pbx.c:2981 ast_register_application: Already have an
application 'MailboxExists'
[Jun 11 22:40:42] WARNING[3230]: pbx.c:2981 ast_register_application: Already have an
application 'VMAuthenticate'

```

Le module semble entrer en conflit avec le module de messagerie déjà chargé. Il faut donc décharger celui-ci avant de charger le module qui nous intéresse.

```
asterisk1*CLI> module unload app_voicemail.so
== Unregistered application 'VoiceMail'
== Unregistered application 'VoiceMailMain'
== Unregistered application 'MailboxExists'
== Unregistered application 'VMAuthenticate'
```

```
asterisk1*CLI> module load app_voicemail_imap.so
== Registered application 'VoiceMail'
== Registered application 'VoiceMailMain'
== Registered application 'MailboxExists'
== Registered application 'VMAuthenticate'
== Parsing '/etc/asterisk/voicemail.conf': Found
== Parsing '/etc/asterisk/users.conf': Found
Loaded app_voicemail_imap.so => (Comedian Mail (Voicemail System) with IMAP
Storage)
```

3 - On configure édite maintenant le fichier voicemail.conf afin de prendre en compte les paramètre IMAP.

```
[general]
imapserver=192.168.20.1 ; Adresse du serveur
imapfolder=inbox      ; Dossier de stockage des messages
imapflags=notls       ; Connexion non chiffrée
authuser=user         ; Nom d'utilisateur
authpassword=user     ; Mot de passe
```

```
[default]
100 => 100,100,,|imapuser=user
101 => 101,101
```

L'extension 100 s'identifiera comme l'utilisateur user sur le serveur IMAP. On peut consulter l'état de la messagerie à l'aide de la commande voicemail show users.

```
asterisk1*CLI> voicemail show users
Context  Mbox User      Zone    NewMsg
default  100 100                0
default  101 101                -1
[Jun 11 21:26:14] WARNING[2847]: app_voicemail_imap.c:8738 mm_log: IMAP Warning:
SECURITY PROBLEM: insecure server advertised AUTH=PLAIN
[Jun 11 21:26:14] WARNING[2847]: app_voicemail_imap.c:2440 messagecount: IMAP
user not set for mailbox 101
```

Les avertissements informent que la connexion est en clair et que l'extension 101 n'est pas présente sur le serveur.

4 - On émet un appel puis on laisse un message et on observe les transactions suivantes :

Time	192.168.0.1	192.168.10.1	192.168.10.2	192.168.20.1
0,000	Request: INVITE sip			SIP/SDP: Request: INVITE sip:100@192.168.10.1
	(5060) -----> (5060)			
0,002	Status: 407 Proxy			SIP: Status: 407 Proxy Authentication Required
	(5060) <----- (5060)			
0,005	Request: ACK sip:10			SIP: Request: ACK sip:100@192.168.10.1
	(5060) -----> (5060)			
0,021	Request: INVITE sip			SIP/SDP: Request: INVITE sip:100@192.168.10.1
	(5060) -----> (5060)			
0,023	Status: 100 Trying			SIP: Status: 100 Trying
	(5060) <----- (5060)			
0,028	Request: INVITE sip			SIP/SDP: Request: INVITE sip:100@192.168.10.2:5072
	(5060) -----> (5072)			
0,033	Status: 100 Trying			SIP: Status: 100 Trying
	(5060) <----- (5072)			
0,303	Status: 100 Trying			SIP: Status: 100 Trying
	(5060) <----- (5073)			
0,497	Status: 180 Ringing			SIP: Status: 180 Ringing
	(5060) <----- (5073)			
0,499	Status: 180 Ringing			SIP: Status: 180 Ringing
	(5060) <----- (5060)			
15,043	Request: CANCEL			SIP: Request: CANCEL sip:100@192.168.10.2:5072
	(5060) -----> (5072)			
15,046	Status: 200 OK			SIP/SDP: Status: 200 OK, with session description
	(5060) <----- (5060)			
15,054	Request: ACK sip:10			SIP: Request: ACK sip:100@192.168.10.1
	(5060) -----> (5060)			
15,098	Status: 200 OK			SIP: Status: 200 OK
	(5060) <----- (5073)			
15,115	Status: 487			SIP: Status: 487 Request Terminated
	(5060) <----- (5073)			
-> Le correspondant ne répond pas, le serveur Asterisk bascule sur la messagerie <-				
15,119	PT=ITU-T G.711 PCMU			RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=24939
	(5076) <----- (14158)			
15,122	Request: ACK			SIP: Request: ACK sip:100@192.168.10.2:5072;transport=udp
	(5060) -----> (5072)			
15,146	PT=ITU-T G.711 PCMU			RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=24940
	(5076) <----- (14158)			
15,168	PT=ITU-T G.711 PCMU			RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=24941
	(5076) <----- (14158)			
25,918	PT=ITU-T G.711 PCMU			RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=25471

```

|(5076) <----- (14158)|      |
[...]
-> Après le « Bip », le serveur Asterisk contact le serveur IMAP <-
|25,942|      | Request: 0000000b |      |IMAP: Request: 0000000b NOOP
|      |      |(52984) -----> (143) |
|25,947|      | Response: 0000000b |      |IMAP: Response: 0000000b OK NOOP completed.
|      |      |(52984) <----- (143) |
|25,951|      | 52984 > imap [ACK] |      |TCP: 52984 > imap [ACK] Seq=16 Ack=30 Win=2800 Len=0
|      |      |(52984) -----> (143) |
|25,953|      | Request: 0000000c |      |IMAP: Request: 0000000c SELECT inbox
|      |      |(52984) -----> (143) |
|25,958|      | Response: * FLAGS |      |IMAP: Response: * FLAGS (\Answered \Flagged \Deleted)
|      |      |(52984) <----- (143) |
|25,965|      | Request: 0000000d |      |IMAP: Request: 0000000d SEARCH UNDELETED UNSEEN HEADER
|      |      |(52984) -----> (143) |
|25,972|      | Response: * SEARCH |      |IMAP: Response: * SEARCH
|      |      |(52984) <----- (143) |
[...]

```

-> L'appelant dépose son message <-

```

|26,065| PT=ITU-T G.711 PCMU |      |      |RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=25473
|(5076) <----- (14158)|      |      |
|26,065| PT=ITU-T G.711 PCMU |      |      |RTP: PT=ITU-T G.711 PCMU, SSRC=0xEA43B983, Seq=8707
|(5076) -----> (14158)|      |      |
|26,078| PT=ITU-T G.711 PCMU |      |      |RTP: PT=ITU-T G.711 PCMU, SSRC=0x5D2FD6D3, Seq=25474
|(5076) <----- (14158)|      |      |
|36,245| PT=ITU-T G.711 PCMU |      |      |RTP: PT=ITU-T G.711 PCMU, SSRC=0xEA43B983, Seq=9215
|(5076) -----> (14158)|      |      |
|36,271| Request: BYE sip:10 |      |      |SIP: Request: BYE sip:100@192.168.10.1
|(5060) -----> (5060) |      |      |
|36,275| Status: 200 OK |      |      |SIP: Status: 200 OK
|(5060) <----- (5060) |      |      |

```

-> Une fois terminé, le serveur Asterix transfère celui-ci vers le serveur IMAP <-

```

|36,530|      | Request: 00000011 |      |IMAP: Request: 00000011 NOOP
|      |      |(52984) -----> (143) |
|36,533|      | Response: 00000011 |      |IMAP: Response: 00000011 OK NOOP completed.
|      |      |(52984) <----- (143) |
|36,534|      | 52984 > imap [ACK] |      |TCP: 52984 > imap [ACK] Seq=228 Ack=802 Win=3336 Len=0

```

```

|      |(52984) -----> (143) |
|36,535|      | Request: 00000012 |      ||IMAP: Request: 00000012 SELECT inbox
|      |(52984) -----> (143) |
|36,538|      | Response: * FLAGS |      ||IMAP: Response: * FLAGS (\Answered \Flagged \Deleted)
|      |(52984) <----- (143) |
|36,541|      | Request: 00000013 A |      ||IMAP: Request: 00000013 APPEND inbox {215891}
|      |(52984) -----> (143) |
|36,555|      | Response: + OK |      ||IMAP: Response: + OK
|      |(52984) <----- (143) |
|36,559|      | Request: Date: Thu |      ||IMAP: Request: Date: Thu, 11 Jun 2009 21:27:25 +0200
|      |(52984) -----> (143) |
[...]
```

5 - On vérifie à nouveau l'état de la messagerie et on peut constater que le message est bien pris en compte.

```

asterisk1*CLI> voicemail show users
Context  Mbox User      Zone  NewMsg
default  100 100                1
default  101 101                -1
```

6 - À l'aide du client de messagerie mutt, on se connecte au serveur IMAP.

```
$ mutt -f imap://user@192.168.20.1/
```

On peut alors voir le message déposé avec en pièce jointe le message vocal.

```

From: Asterisk PBX <asterisk@asterisk1>
To: 100 <user>
Subject: [PBX]: New message 6 in mailbox 100
X-Asterisk-VM-Message-Num: 6
X-Asterisk-VM-Server-Name:
X-Asterisk-VM-Context: default
X-Asterisk-VM-Extension: 100
X-Asterisk-VM-Priority: 2
X-Asterisk-VM-Caller-channel: SIP/101-0a08e640
X-Asterisk-VM-Caller-ID-Num: 101
X-Asterisk-VM-Caller-ID-Name: (null)
X-Asterisk-VM-Duration: 9
X-Asterisk-VM-Orig-date: Thursday, June 11, 2009 at 09:27:25 PM
```

X-Asterisk-VM-Orig-time: 1244748445  
X-Asterisk-CallerID: 101

[-- Attachement #1 --]  
[-- Type : text/plain, Codage : 8bit, Taille : 0,2K --]

Dear 100:

Just wanted to let you know you were just left a 0:09 long message (number 6) in mailbox 100 from 101, on Thursday, June 11, 2009 at 09:27:25 PM so you might want to check it when you get a chance. Thanks!

--Asterisk

[-- Attachement #2: Voicemail sound attachment. --]  
[-- Type : audio/x-wav, Codage : base64, Taille : 206K --]

[-- audio/x-wav n'est pas disponible (utilisez 'v' pour voir cette partie) --]

7 - Le message est maintenant marqué comme lu, vérifions sont état dans Asterisk :

```
asterisk1*CLI> voicemail show users
Context  Mbox  User      Zone    NewMsg
default  100  100              0
default  101  101              -1
```

## TP 4 – Plan de numérotation : configuration avancée.

### 1. Applications

1- Configurez les applications suivantes et testez-les:

#### MeetMe : Conférence

Exemple d'appel à une conférence statique

Dans meetme.conf:

```
conf => 900  
conf => 910,1234
```

La conférence 900 étant une conférence publique, tandis que la conférence 910 est une conférence privée avec un Code PIN : 1234.

Dans extension.conf:

```
exten=> 900,1,Answer()  
exten=> 900,2,MeetMe(900)  
(idem pour 910)
```

Quand un utilisateur compose l'extension 900, Asterisk le place automatiquement dans la conférence 900.

Si un autre utilisateur fait de même, il entrera dans la conférence 900 et, les deux utilisateurs pourront alors entrer en communication. Et ainsi de suite.

#### Parked Call

Dans features.conf:

```
[general]  
parkext => 8000  
parkpos => 8001-8010 ;ainsi on pourra « parké » (du verbe parker venant de parking) 10 personnes  
context => parkedcalls
```

Dans extensions.conf:

```
[moncontexte]  
include=> parkedcalls  
exten=> 8001,1,ParkedCall(8001)  
exten=> 8001,hint,park:8001@parkedcalls
```

#### MusicOnHold

```
[default]  
mode=files
```

```
directory=/var/lib/asterisk/moh
```

Ensuite il faut redémarrer l'appli, puis créer un fichier mp3 et pour le convertir sous plusieurs codecs. Pour finir on reloadera les modules.

```
file convert music.mp3 music.alaw
file convert music.mp3 music.gsm
reload
```

Avec 3 terminaux, lorsque l'on mets en attente **avec le même** téléphone les deux autres, nous avons le **même** flux audio (la musique ne redémarre pas sur le deuxième téléphone mis en attente). Or lorsque chacun des téléphones mettent en attente les 2 autres, le flux audio redémarre.

## 2. File d'attentes et agents

1- Configurez le fichier queues.conf et créez une nouvelle file d'attente :

```
[MaQueue]
music=default
strategy=ringall
timeout=15
retry=5
wrapuptime=0
maxlen=0
announce-frequency=0
announce-holdtime=no
```

2- Ajoutez des agents dans le fichier agents.conf

```
ackcall=no
```

```
agent => 1001,4321,AgentAlpha
agent => 1002,5432,AgentBravo
agent => 1003,6543,AgentCharlie
```

Puis a la suite de [MaQueue] dans queues.conf

```
member=> Agent/1001
member=> Agent/1002
member=> Agent/1003
```

Log d'agents :

```
Agent Login
exten=> 1001,1,AgentCallbackLogin(||${CALLERIDNUM}@moncontexte)
Agent Logout
exten=> 1002,1,AgentCallbackLogin(|||)
```

### 3. ACD : Menus vocaux

1- Concevez un menu ayant comme la structure suivante :

1=> Appelle un numérotation

2=> Joue une musique d'attente 20 seconde, puis appelle la boite vocale

3=> Accède à une salle de conférence

4=> Mets l'appel dans un parc

5=> Enregistre le numéro de l'appelant dans la variable lastcall de la base de donnée puis raccroche.

Dans extensions.conf :

[mainmenu]

exten => s,1,Answer

exten => s,2,Background(annonce)

exten => s,3,DigitTimeout,5

exten => s,4,ResponseTimeout,10

; 1- Appelle le 300

exten => 1,1,Dial(SIP/300)

; 2- Joue la MusicOnHold 20sec puis Voicemail

exten => 2,1,PlayBack(toto20sec.mp3)

exten => 2,1,VoicemailMain()

exten => 2,2,Hangup

; 3- Accéder à la salle de conférence

exten => 3,1,Dial(SIP/900)

; 4- Accéder au Park

exten => 4,1,Goto([8000@moncontexte](#))

; 5- Enregistre le numéro de l'appelant

exten => 5,1,Set(DB(lastcall/caller)=\${CALLERID})

[default]

include => mainmenu

## TP 5 – Communication Inter Serveurs.

### 1. Communication inter serveur avec SIP

1 - Nous allons maintenant configurer 2 serveurs Asterisk de manière à ce qu'ils communiquent entre eux via le protocole SIP et ainsi pouvoir effectuer des appels entre les correspondants présents au niveau de chaque serveur Asterisk. Pour cela, on modifie les fichiers sip.conf et extensions.conf de la façon suivante.

## **Asterisk 1** :

### **sip.conf**

; Configuration générale

[general]

context=default ; Context par défaut

allowoverlap=no ; Ne pas autoriser le recouvrement de numéros

bindport=5060 ; Port à écouter pour la signalisation

bindaddr=0.0.0.0 ; Adresse IP à écouter (0.0.0.0 pour toute)

srvlookup=no ; Activer la résolution DNS

[ast2]

type=friend

host=192.168.20.1

fromuser=ast1

context=ast1

; Compte 100

[100]

type=friend ; Type du compte

callerid=100

secret=100 ; Mot de passe

context=local ; Context

host=dynamic ; Hôte dynamique, force l'enregistrement

; Compte 101

[101]

type=friend ; Type du compte

callerid=101

secret=101

context=local ; Context

host=dynamic

### **extensions.conf**

[local]

include => ast1

include => ast2

include => voicemail

[ast1]

exten => \_1XX,1,Dial(SIP/\${EXTEN}, 15)

exten => \_1XX,n,VoiceMail(\${EXTEN}@default, u)

exten => \_1XX,n,Hangup()

[ast2]

exten => \_2XX,1,Dial(SIP/ast1/\${EXTEN}, 15)

[voicemail]

exten => 700,1,VoiceMailMain()

## **Asterisk 2** :

### **sip.conf**

; Configuration générale

[general]

context=default ; Context par défaut  
allowoverlap=no ; Ne pas autoriser le recouvrement de numéros

bindport=5060 ; Port à écouter pour la signalisation  
bindaddr=0.0.0.0 ; Adresse IP à écouter (0.0.0.0 pour toute)  
srvlookup=no ; Activer la résolution DNS

[ast1]

type=friend ; Type du compte  
host=192.168.10.1 ; Adresse IP du serveur  
fromuser=ast2 ; En provenance de...  
context=ast2

; Compte 200

[200]

type=friend ; Type du compte  
callerid=200  
secret=200 ; Mot de passe  
context=local ; Context  
host=dynamic ; Hôte dynamique, force l'enregistrement

; Compte 201

[201]

type=friend ; Type du compte  
callerid=201  
secret=201  
context=local ; Context  
host=dynamic

### **extensions.conf**

[local]

include => ast1  
include => ast2  
include => voicemail

[ast2]

exten => \_2XX,1,Dial(SIP/, 15)  
exten => \_2XX,n,VoiceMail(@default, u)  
exten => \_2XX,n,Hangup()

[ast1]

exten => \_1XX,1,Dial(SIP/ast1/\${EXTEN})

[voicemail]

exten => 700,1,VoiceMailMain()

2 - On peut maintenant vérifier que chacun s'est correctement enregistré auprès de l'autre

avec la commande sip show peers :

### **Asterisk 1**

```
asterisk1*CLI> sip show peers
```

Name/username	Host	Dyn	Nat	ACL	Port	Status
101/101	(Unspecified)	D		0		Unmonitored
100/100	192.168.10.2	D		5072		Unmonitored
ast2	192.168.20.1			5060		Unmonitored

3 sip peers [Monitored: 0 online, 0 offline Unmonitored: 2 online, 1 offline]

### **Asterisk 2**

```
asterisk2*CLI> sip show peers
```

Name/username	Host	Dyn	Nat	ACL	Port	Status
201/201	192.168.10.254	D		5060		Unmonitored
200/200	192.168.20.2	D		5068		Unmonitored
ast1	192.168.10.1			5060		Unmonitored

3 sip peers [Monitored: 0 online, 0 offline Unmonitored: 3 online, 0 offline]

3 - Comme on peut le voir, les serveurs *Asterisk 1* et *Asterisk 2* sont respectivement enregistrés sous les noms ast1 et ast2 l'un auprès de l'autre.