

# Compte-rendu TP3

---

- Compte de rendu de :
  - Chhiny LIM
  - Amine ABDOUL-AZID
- Efrei NSM1 : Networks and Security Manager
- Salim BENAYOUNE : Module VOIP



## réalisations

---

### Question 1 :

Sur le serveur A : trunksip .153

- Sur la VM A : *sip.conf* :

```
[general]
context=public
bindaddr=0.0.0.0
transport=udp

[amine]
type=friend
callerid="My name" <100>
host=dynamic
secret=test
context=internal

[trunk_A_vers_B]
type=friend
secret=azerty
context=internal
host=dynamic
insecure=port,invite
```

- Sur la VM A : *extensions.conf* :

```
[internal]
exten => 600,1,Playback(demo-echotest)
```

```

exten => 600,n,Echo

exten => 100,1,Dial(SIP/amine)
exten => 200,1,Dial(SIP/chhiny)

exten => 900,1,Dial(SIP/trunk_A_vers_B/${EXTEN})

```

## Sur le serveur B : peersipe .154

- Sur la VM B : *sip.conf* :

```

[general]
context=internal
bindaddr=0.0.0.0
transport=udp
register => trunk_A_vers_B:azerty@192.168.1.153

[chhinyboss]
type=friend
callerid="My name" <900>
host=dynamic
secret=vitrygtr
context=internal

```

- Sur la VM B : *extensions.conf* :

```

[internal]

exten => 600,1,Playback(demo-echotest)
exten => 600,n,Echo

exten => 100,1,Dial(SIP/amine)
exten => 900,1,Dial(SIP/chhinyboss)

```

## Récapitulatif de la question 1

- On Précise sur la VMA :
  - Le contexte (internal)
  - Un mot de passe
  - On relève son adresse IP
  - Dans le *extensions.conf* : On précise les numéros de téléphone et `900,1,Dial(SIP/trunk_A_vers_B/${EXTEN})` la plage de début.
- On précise sur la VMB :
  - On précise le lien trunk dans la section register sous le format : `register => $user_trunk:$mdp@$ip_trunk_vm_a`

- On précise dans le *extensions.conf*
  - Les numéros connus.

## Question 2 :

### 1. SIP INVITE

- Voici un paquet **SIP INVITE** un ACK a lieu entre **chhinyaleboss** et **amine**.

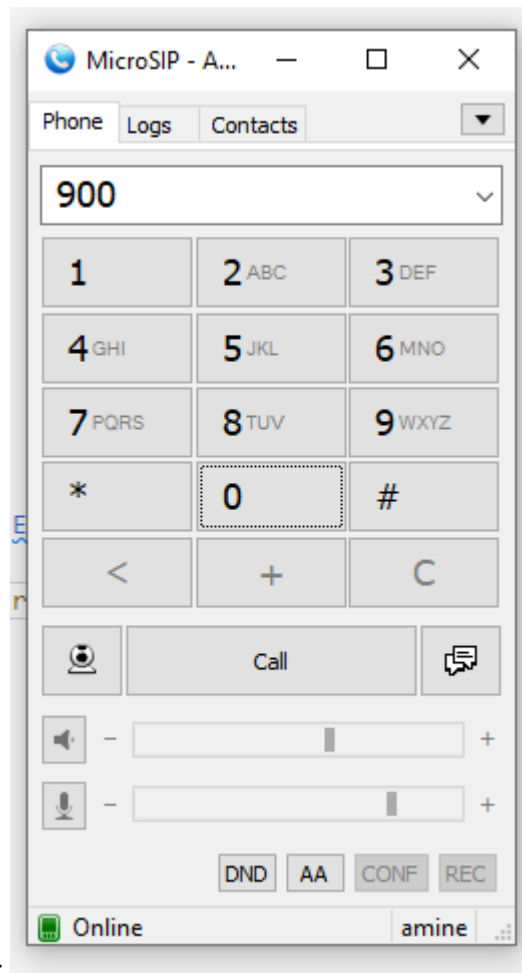
106	10.325694807	94:08:53:bd:29:4f	PcsCompu 53:54:e6	ARP	60	192.168.1.110	is at 94:08:53:bd:29:4f
107	10.511283418	192.168.1.153	192.168.1.154	SIP/SDP	959	Request: INVITE sip:900@192.168.1.154:5060, in-dialog	
108	10.511980524	192.168.1.154	192.168.1.153	SIP	591	Status: 100 Trying	
109	10.512218059	192.168.1.154	192.168.1.153	SIP/SDP	889	Status: 200 OK	
110	10.512694952	192.168.1.154	192.168.1.110	SIP/SDP	958	Request: INVITE sip:chhinyaleboss@192.168.1.110:65015;ob, in-dialog	
111	10.513105290	192.168.1.153	192.168.1.154	SIP	450	Request: ACK sip:900@192.168.1.154:5060	
112	10.513142800	192.168.1.153	192.168.1.154	SIP	484	Request: BYE sip:900@192.168.1.154:5060	
113	10.513870806	192.168.1.154	192.168.1.153	SIP	508	Status: 200 OK	
114	10.514137927	192.168.1.110	192.168.1.154	SIP/SDP	944	Status: 200 OK	
115	10.514787153	192.168.1.154	192.168.1.110	SIP	498	Request: ACK sip:chhinyaleboss@192.168.1.110:65015;ob	
116	10.515666174	192.168.1.154	192.168.1.110	SIP/SDP	958	Request: INVITE sip:chhinyaleboss@192.168.1.110:65015;ob, in-dialog	
117	10.517165928	192.168.1.110	192.168.1.154	RTCP	102	Receiver Report Source description	
118	10.517384888	192.168.1.110	192.168.1.154	SIP/SDP	944	Status: 200 OK	
119	10.518436848	192.168.1.154	192.168.1.110	SIP	498	Request: ACK sip:chhinyaleboss@192.168.1.110:65015;ob	
120	10.519219221	192.168.1.154	192.168.1.110	SIP	532	Request: BYE sip:chhinyaleboss@192.168.1.110:65015;ob	
121	10.519748190	192.168.1.110	192.168.1.154	SIP	374	Status: 200 OK	
122	10.519985298	192.168.1.110	192.168.1.154	RTCP	110	Receiver Report Source description Goodbye	

[capture1\\_question2et3.pcapng](#)

## Question 3 :

- Pour faire apparaître le maximum des messages SIP et les erreurs associées, on va faire des manipulations sur les messages SIP INVITE.
  - On initie le dialogue avec le client1 et on lui envoie un INVITE.
  - On initialise le dialogue avec le client2 et on lui envoie un INVITE.
  - On initialise le dialogue avec le client1 et on lui envoie un ACK.
  - On initie la requête SIP mais on n'accepte pas l'appel afin de provoquer une erreur.
  - On initie la requête SIP mais on coupe l'appel afin de provoquer une erreur.

## Question 4 :



- J'appuie sur la webcam sur **microsip** :

[capture3\\_question4\\_video.pcapng](#)

- RTCP :
  - Les échanges RTP/RTCP sont réalisés à travers le protocole RTP. Le codec G.711 est utilisé, le SSRC est unique pour chaque échange.
  - Nous le constatons durant les échanges vidéos.

87	19.782116241	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9248, Time=2560
88	19.801691350	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32063, Time=2720
89	19.801901866	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9249, Time=2720
90	19.821639570	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32064, Time=2880
91	19.821828336	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9250, Time=2880
92	19.842188355	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32065, Time=3040
93	19.842414962	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9251, Time=3040
94	19.861353374	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32066, Time=3200
95	19.861538277	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9252, Time=3200
96	19.881478465	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32067, Time=3360
97	19.881670822	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9253, Time=3360
98	19.901304511	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32068, Time=3520
99	19.901460633	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9254, Time=3520
100	19.921667033	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32069, Time=3680
101	19.921816112	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9255, Time=3680
102	19.941539481	192.168.1.153	192.168.1.154	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x51EB13D0, Seq=32070, Time=3840
103	19.941779588	192.168.1.154	192.168.1.110	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x5A018B35, Seq=9256, Time=3840

- Pour maximiser la transmission de la vidéo, on utiliserait le codec H.264.

## Partie 2 : Services téléphoniques

### Question 1 : Transfert d'appels

- Nous avons le fichier *extensions.conf* suivant :

```
GNU nano 5.4 /etc/asterisk/extensions.conf

[globals]

[general]
autofallthrough=yes

[default]

[incoming_calls]

[phones]
include => internal
include => remote

[internal]
exten => _2XXX,1,NoOp()
exten => _2XXX,n,Dial(SIP/${EXTEN},30,tT)
exten => _2XXX,n,Playback(the-party-you-are-calling&is-curntly-unavail)
exten => _2XXX,n,Hangup()

[remote]
exten => _1XXX,1,NoOp()
exten => _1XXX,n,Dial(SIP/asterisk2/${EXTEN})
exten => _1XXX,n,Hangup()

[asterisk2_incoming]
include => internal
```

Pour transférer un appel : nous allons mettre en place le blind transfer

Lorsque nous utilisons **la touche dièse #** , nous verrons que l'appel sera transféré vers le numéro suivant.

Pour cela nous activons dans le fichier *features.conf* : dans la section [featuremap] on ajoute :

```
[featuremap]
blindxfer => #1
atxfer => *2
```

```
GNU nano 5.4 /etc/asterisk/features.conf

;atxferdropcall = no ; If someone does an attended transfer, then hangs up before the transfer
; ; target answers, then by default, the system will try to call back the
; ; person that did the transfer. If this is set to "yes", the ringing
; ; transfer target is immediately transferred to the transferee.
;atxferloopdelay = 10 ; Number of seconds to sleep between retries (if atxferdropcall = no)
;atxfercallbackretries = 2 ; Number of times to attempt to send the call back to the transferer.
; ; By default, this is 2.
;transferdialattempts = 3 ; Number of times that a transferer may attempt to dial an extension before
; ; being kicked back to the original call.
;transferretrysound = beep ; Sound to play when a transferer fails to dial a valid extension.
;transferinvalidsound = beeperr ; Sound to play when a transferer fails to dial a valid extension and is out of retries.
;atxferabort = *1 ; cancel the attended transfer
;atxfercomplete = *2 ; complete the attended transfer, dropping out of the call
;atxferthreeway = *3 ; complete the attended transfer, but stay in the call. This will turn the call into a multi-party bridge
;atxferswap = *4 ; swap to the other party. Once an attended transfer has begun, this options may be used multiple times

; Note that the DTMF features listed below only work when two channels have answered and are bridged together.
; They can not be used while the remote party is ringing or in progress. If you require this feature you can use
; chan_local in combination with Answer to accomplish it.

[featuremap]
;blindxfer => #1 ; Blind transfer (default is #) -- Make sure to set the T and/or t option in the Dial() or Queue() app call!
;disconnect => *0 ; Disconnect (default is *) -- Make sure to set the H and/or h option in the Dial() or Queue() app call!
;automon => *1 ; One Touch Record a.k.a. Touch Monitor -- Make sure to set the W and/or w option in the Dial() or Queue() app call!
;atxfer => *2 ; Attended transfer -- Make sure to set the T and/or t option in the Dial() or Queue() app call!
;parkcall => #72 ; Park call (one step parking) -- Make sure to set the K and/or k option in the Dial() app call!
;automixmon => *3 ; One Touch Record a.k.a. Touch MixMonitor -- Make sure to set the X and/or x option in the Dial() or Queue() app call!

[applicationmap]
; Note that the DYNAMIC_FEATURES channel variable must be set to use the features
; defined here. The value of DYNAMIC_FEATURES should be the names of the features
; to allow the channel to use separated by '#'. For example:
; Set(__DYNAMIC_FEATURES=mvfeature1#mvfeature2#mvfeature3)
```

- Ensuite on recharge la configuration...

Question 2 : Messagerie vocale et consultation de la messagerie vocale

## Mise en place de la messagerie vocale

- La commande Record() est utilisée pour enregistrer des messages vocaux. Le premier message est dédiée pour l'auto-attendant et le second pour l'IVR :
- Nous modifions la section [from-internal] dans le le fichier *extensions.conf* :

```
[from-internal]
exten => _4.,1,Record(${EXTEN:1}:gsm)
exten => _4.,n,wait(1)
exten => _4.,n,Playback(${EXTEN:1})
exten => _4.,n,Hangup()
```

- En fait il enregistre un séquence audio spous unn format ici gsm
- A vrai dire le **4.** signifie que on accepte tout autre caractère suivi du n°4.
- On appelle le numéro 4 pour enregistrer un message vocal. **4menu1**

"Merci de bien vouloir presser une touche ou d'attendre votre tour".

- On presse la touche # pour le réécouter.
- On répète les étapes mais pour l'IVR : avec 4menu2

"Presser 1 pour l'accueil, 2 pour le rayon informatique et 3 pour le rayon vêtement".

## Mise en place pour consulter de la messagerie vocale

- La consultation de la messagerie vocale :
- Nous ajoutons dans le fichier **voicemail.conf** :

```
[general]
format=wav49|gsm|wav
[default]
100=>100,Caixa do PAP2,root@localhost,,|attach=yes|delete=0
900=>900, chhiny, root@localhost,,|attach=yes|delete=0
```

- Nous ajoutons dans le fichier *extensions.conf* :

```
[stdexten]
exten=>s,1,Dial(${ARG1},20,tT)
exten=>s,n,Goto(${DIALSTATUS})
exten=>s,n,hangup()
exten=>s,n(BUSY),voicemail(${ARG2},b)
exten=>s,n,hangup()
exten=>s,n(NOANSWER),voicemail(${ARG2},u)
exten=>s,n,hangup()
exten=>s,n(CANCEL),hangup
```

```

exten=>s,n(CHANUNAVAIL),hangup
exten=>s,n(CONGESTION),hangup

[from-internal]
exten=>100,1,Gosub(stdexten,s,1(SIP/amine,${EXTEN}))
exten=>900,1,Gosub(stdexten,s,1(SIP/chhiny,${EXTEN}))
exten=9,1,voicemailmain()

```

- Nous testons en appuyant sur la **touche 9**. Afin de **consulter la messagerie vocale**.
- Par conséquent nous avons réadapté le fichier *extensions.conf* :
  - Il attends 20 secondes avant de passer à la messagerie vocale. `s,1,Dial(${ARG1},20,tT)`
  - On peut changer de contexte grâce au Goto et ainsi récupère la valeur de DIALSTATUS.
  - Elle correspond à la réponse par exemple n'a pas répondu car il est déjà en communication ou le téléphone est éteint ou voire si le numéro n'est plus attribué.
  - Si il n'a pas de réponse `exten=>s,n(NOANSWER),voicemail(${ARG2},u)`, on lui laisse un message.
  - Si le canal, n'est pas disponible ou que la personne refuse volontairement l'appel exemple l'arnaque au CPF... `exten=>s,n(CHANUNAVAIL),hangup` et `exten=>s,n(CANCEL),hangup`

### Question 3 : Standard automatique

- Nous mettons en place une auto-attendant aussi connu sous le nom de *standard automatique*.
- Mais également en tant que IVR :
- En effet IVR signifiant *Interactive Voice Response system*.
- Par conséquent, un Serveur Vocal Interactif est un système automatique qui permet de dialoguer avec l'appelant afin de déterminer le plus finement possible le motif de son appel.

#### Mise en place avec la méthode auto réceptionniste :

- Mise en place avec la méthode auto réceptionniste dans le fichier *extensions.conf* :

```

[from-internal]
exten=>8,1,goto(trunk_A_vers_B,9999,1)
[from-siptrunk]
include=trunk_A_vers_B
[trunk_A_vers_B]
exten=>9999,1,answer()
exten=>9999,n,background(menu1)
exten=>9999,n,waitexten(10)
exten=>9999,n,Dial(${OPERATOR})
exten=>100,1,Dial(SIP/amine)
exten=>900,1,Dial(SIP/chhiny)

```

- Nous appelons **le 8** et nous testons en appuyant sur le **900** afin d'être redirigé vers le **SIP/chhiny**.
- Nous allons vers la **priorité n°1**.

- **Waitextension** est une fonction qui permet de faire attendre l'appelant pendant un certain temps.
- Il fera appel à **OPERATOR** pour le rediriger vers le **SIP/chhiny** par exemple.

### Mise en place avec la méthode IVR

- Dans le fichier *extensions.conf* nous ajoutons :

```
[from-siptrunk]
include=ivr sip
[from-internal]
exten=>8,1,goto(ivr sip,9999,1)
[ivr sip]
exten=>9999,1,answer()
exten=>9999,n,background(menu2)
exten=>9999,n,waitexten(10)
exten=>9999,n,Dial(${OPERATOR})
exten=>1,1,dial(SIP/amine)
exten=>2,1,dial(SIP/chhiny)
exten=>3,1,dial(IAX/amine2)
exten=>100,1,Dial(SIP/amine)
exten=>900,1,Dial(SIP/chhiny)
```

- Ensuite nous appelons **le 8** afin de tester son fonctionnement et de choisir les différentes options.
- Il cherche l'extension 9999 dans le contexte *ivr sip* en gros c'est une étiquette pour se brancher.
- Le menu2 est utilisé toujours avec le préfixe 4. soit 4menu2.
- On tape 1 pour amine , 2 pour chhiny et 3 pour amine2.

### Question 4 : Conférence

- La conférence est un service qui permet de communiquer entre plusieurs personnes.
- En effet cette dernière est assez simple à mettre en place et dans notre contexte, on peut la mettre en place en utilisant un appel entre trois personnes.
- Dans le fichier *extensions.conf*, on peut trouver la ligne suivante :

```
exten=4,1,Confbridge(main)
```

- Cette directive **Confbridge(main)** permet de démarrer une conférence.
- Le **numéro 4** lors de l'appel permettra de démarrer la conférence. En effet ce numéro est souvent utilisé avec la plupart des softphones.
- On peut aussi la spécifier de cette manière :



```
exten => 1,1,Answer()
exten => 1,n,ConfBridge(1234,,1234_participants,1234_menu)
```

## confbridge

### Question 5 : Interception dans le groupe

- Introduction :
  - Le parage d'appels permet à une personne de mettre un appel en attente sur un poste téléphonique et de poursuivre la conversation à partir de n'importe quel autre poste téléphonique.
- 2 modes sont possibles : "Park Pickup Config" ou "Directed Call Pickup"
  - "Park Pickup Config" :
    - on appelle le numéro de l'appelant
    - on attend qu'un poste téléphonique appelle le numéro de l'appelé
    - on récupère l'appel
    - on continue la conversation
- Nous pouvons aussi nous intéresser à ces méthodes :
  - Call Pickup reprendre un appel
  - FOLLOW ME : Pour continuer à suivre une fonctionnalité
  - Cela nécessite l'ajout d'un troisième poste téléphonique.

### Mise en place du call parking

Le **call parking** s'active dans le fichier *extensions.conf*

```
[from-internal]
include => parkedcalls
```

- Cela peut aussi être fait de la manière suivante : dans le fichier *res\_parking.conf*.

```
[general]
[default]
parkext => 700
parkpos => 701-720
context => parkedcalls
```

### Mise en place du call pickup

- Tandis que le **call pickup** s'active dans le fichier *features.conf* : et en modifiant ensuite le fichier *extensions.conf*

```
pickupexten = *8
```

- Pour recharger la modification : du fichier *features.conf* dans le terminal :  

```
module reload features
```
- Ensuite on modifie le fichier *extensions.conf* :

```
callgroup=1  
pickupgroup=1  
directmedia=no
```

- Pour recharger la modification : du fichier *extension.conf* dans le terminal :  

```
sip reload
```

## CALLPARK

### Question 6 : Un petit call center

- Un call parking est nécessaire pour un call center.
- Un agent avec des queues peut être utilisé pour gérer les appels entrants. Et le trafic dans un call center.
- En effet nous voulons créer un call center via Asterisk : En général cela s'appelle Automated call distributor : ACD
- Nous avons aussi besoin d'une messagerie vocale : *voicemail.conf* que nous avons défini précédemment.  
[Using queues.conf](#)
- Voici le fichier *queues.conf* :

```
[general]  
persistentmembers = yes  
  
; General sales queue  
[sales-general]  
context=sales  
music=default  
strategy=ringall  
joinempty=strict  
leavewhenempty=strict  
; Customer service queue  
[customerservice]  
context=customerservice  
music=default  
strategy=ringall  
joinempty=strict  
leavewhenempty=strict
```

- Mais on peut simplifier le processus d'autres modules que nous avons déjà vu.
- De cette manière : aussi on peut le définir dans le *queues.conf* :

```
[general]
autofill=yes
shared_lastcall=yes
[StandardQueue](!)
musicclass=default
strategy=rrmemory
joinempty=no
leavewhenempty=yes
ringinuse=no
[sales](StandardQueue)
[support](StandardQueue)
```

- Le fichier *extensions.conf* sera réadapté : afin d'ajouter la partie dédiée à l'ACD : Automated Distributed Call Center

```
[Queues]
exten => 7001,1,Verbose(2,${CALLERID(all)} entering the support queue)
same => n,Queue(support)
same => n,Hangup()

exten => 7002,1,Verbose(2,${CALLERID(all)} entering the sales queue)
same => n,Queue(sales)
same => n,Hangup()

[LocalSets]
include => Queues ; allow phones to call queues
```

- Sources :
  - [ACR\\_asterisk](#)
  - [voip-info.org](#)
  - [ACD](#)
  - [asterisk\\_book\\_chunk\\_acd](#)

## Les fichiers de configuration final :

### **sip.conf**

```
[general]
bindport=5060
bindaddr=0.0.0.0
context=dummy
disallow=all
allow=ulaw
```

```

alwaysauthreject=yes
allowguest=no

register=1040:vitrygtr@sip.kaiba.corp:5600/9999
register => trunk_A_vers_B:azerty@192.168.1.153

[group1](!)
type=friend
secret=vitrygtr
host=dynamic
qualify=yes
callgroup=1
pickupgroup=1
directmedia=no
context=from-internal

[amine](group1)
[chhiny](group1)
[kaiba](group1)
[makuba](group1)

[siptrunk]
type=peer
defaultuser=1040
secret=vitrygtr
port=5600
insecure=invite
host=sip.kaiba.corp
fromuser=1040
fromdomain=sip.kaiba.corp
context=from-siptrunk

```

## extensions.conf

```

[globals]
OPERATOR=SIP/chhiny

[from-internal]
include=>parkedcalls

exten => _4.,1,Record(${EXTEN:1}:gsm)
exten => _4.,n,wait(1)
exten => _4.,n,Playback(${EXTEN:1})
exten => _4.,n,Hangup()

exten=>900,1,Gosub(stdexten,s,1(SIP/amine,${EXTEN}))
exten=>6002,1,Gosub(stdexten,s,1(SIP/chhiny,${EXTEN}))
exten=>6003,1,Gosub(stdexten,s,1(SIP/makuba,${EXTEN}))
exten=>6004,1,Gosub(stdexten,s,1(SIP/kaiba,${EXTEN}))

```

```

exten=>_9.,1,dial(SIP/siptrunk/${EXTEN:1},20)

exten=6,1,Confbridge(main)
exten=7,1,goto(trunk_A_vers_B,9999,1)
exten=9,1,voicemailmain()

exten => 8100,1,Answer()
exten => 8100,n,MusicOnHold(default,30)

[from-siptrunk]
include=trunk_A_vers_B

[trunk_A_vers_B]
exten=>9999,1,answer()
exten=>9999,n,background(menu2)
exten=>9999,n,WaitExten(10)
exten=>9999,n,Dial(${OPERATOR})
exten=>1,1,dial(SIP/amine)
exten=>2,1,dial(SIP/chhiny)
exten=>3,1,dial(SIP/kaiba)
exten=>100,1,Dial(SIP/amine)
exten=>900,1,Dial(SIP/chhiny)

[stdexten]
exten=>s,1,Dial(${ARG1},20,tT)
exten=>s,n,FollowMe(${ARG2})
exten=>s,n,Goto(${DIALSTATUS})
exten=>s,n,hangup()
exten=>s,n(BUSY),voicemail(${ARG2},b)
exten=>s,n,hangup()
exten=>s,n(NOANSWER),voicemail(${ARG2},u)
exten=>s,n,hangup()
exten=>s,n(CANCEL),hangup
exten=>s,n(CHANUNAVAIL),hangup
exten=>s,n(CONGESTION),hangup

```

## features.conf

### res\_parking.conf :

```

[general]
[default]                                ; Default Parking Lot
parkext => 700                            ; What extension to dial to park. (optional; if
parkpos => 701-720                        ; What range of parking spaces to use - must be
numeric
context => parkedcalls                    ; Which context parked calls and the default park

```