


```

exten => _1234567890,1,Verbose(Incoming call on Messagenet)
exten => _1234567890,n,Dial(SIP/321,45)
exten => _1234567890,n,Hangup()

; Make the "321" internal available to other from-sip callers
exten => _321,1,Dial(SIP/321,45)
exten => _321,n,Hangup()

;;; Allow internals to call any Messagenet internal number ;;;;;;;;;;;;;;

exten => _5XXXXXX,1,Dial(SIP/sip.messagenet.it/${EXTEN})

;;; Example outbound for Italy ;;;;;;;;;;;;;;

; Route International calls ("00" prefix) on Messagenet
exten => _00Z.,1,Macro(messagenet-dial)

; Route National landline calls on Messagenet
exten => _0Z.,1,Macro(messagenet-dial)

; Route National mobile calls on Messagenet
exten => _3.,1,Macro(messagenet-dial)

; Route FreePhone/special calls on Messagenet
exten => _80[03].,1,Macro(messagenet-dial)

exten => _84[0178].,1,Macro(messagenet-dial)

; Route service calls on Messagenet
exten => _199.,1,Macro(messagenet-dial)

[macro-messagenet-dial]
exten => s,1,Dial(SIP/sip.messagenet.it/${MACRO_EXTEN})
exten => s,n,Hangup()

```

Esempio di configurazione per il demultiplexing sulla base dell'header SIP "To"

Come fare a discriminare le chiamate in ingresso sulla base del numero chiamato, qualora queste chiamate fossero sulla stessa URI. Forse così descritto il quadro è un po' fumoso, quindi facciamo un esempio fittizio.

Il cliente ha la URI 5XXXXX, sulla quale sono diretti due numeri, il numero di Roma 0612345678 e il numero di Milano 0212345678. L'idea è che quindi potrà in linea di massima ricevere chiamate dirette a:

- Il numero di Roma: 0612345678
- Il numero di Milano: 0212345678
- Il numero interno: 5XXXXX

Innanzitutto, nel file *sip.conf*, come specificato dai nostri esempi di configurazione, avrà scritto:

```

[general]
port=5060
bindaddr=0.0.0.0
context=sipin
nat=yes
disallow=all
allow=alaw
; allow=g729
; allow=ilbc
realm=sip.messagenet.it
rtptimeout=60
rtpholdtimeout=300
; incoming calls
register => 5312345:password@sip.messagenet.it:5061/200

```

Prestiamo molta attenzione alla riga che comincia con *register*: dopo le credenziali di accesso, c'è riportato un */200*. Quello che il cliente mette dopo lo */*, è l'estensione locale a cui verranno consegnate le chiamate in ingresso su questa registrazione.

Per questo, dunque, è necessario che nel dialplan sia configurata un'estensione 200, e che questa operi in modo che le chiamate vengano discriminate in base al numero di destinazione, e quindi trattate in modi diversi. Un esempio di come farlo è il seguente:

```

[sipin]
exten => 200,1,Set(CALLED=${CUT(CUT(SIP_HEADER(TO),@,1),:,2)})
exten => 200,n,Verbose(Chiamata su Messagenet per il numero ${CALLED})
exten => 200,n,GotoIf("${CALLED}" == "0612345678"?to-rome,s,1)
exten => 200,n,GotoIf("${CALLED}" == "0212345678"?to-milan,s,1)
exten => 200,n,GotoIf("${CALLED}" == "5312345"?to-internal,s,1)
exten => 200,n,Goto(to-other,s,1)

[to-rome]
exten => s,1,Verbose(Chiamata per Roma)
exten => s,n,Dial(SIP/201) exten => s,n,Hangup()
[to-milan]
exten => s,1,Verbose(Chiamata per Milano)
exten => s,n,Dial(SIP/202) exten => s,n,Hangup()
[to-internal]

```

```
exten => s,1,Verbose(Chiamata per l'interno Messagenet)
exten => s,n,Dial(SIP/203) exten => s,n,Hangup()
[to-other] exten => s,1,Verbose(Chiamata per numero non configurato)
exten => s,n,Playback(pbx-invalid) exten => s,n,Hangup()
Ecco cosa accade in questo dialplan di esempio:
```

- La chiamata arriva, e viene diretta nel context sipin, sull'extension 200.
- La variabile *CALLED* viene riempita con una porzione dell'header SIP, quella che dovrebbe contenere il numero chiamato.
- Vengono fatti sequenzialmente dei controlli sulla variabile *CALLED*, durante i quali, se il numero in essa contenuto risulta essere uno di quelli elencati, si viene inviati ad un nuovo context opportuno.

E i context sono così descritti:

- *to-rome* gira la chiamata all'extension locale 201.
- *to-milan* gira la chiamata all'extension locale 202.
- *to-internal* gira la chiamata all'extension locale 203.
- *to-other* riproduce un messaggio vocale che cita che l'extension non è valida, quindi riappende.

In questo dialplan sono presenti una serie di comandi *Verbose* che possono essere commentati, ma che fanno in modo di poter seguire una chiamata da una console di Asterisk per verificare che si comporti come desiderato.

Il risultato ottenuto, ovviamente, è che a seconda del numero chiamato, le chiamate vengono dirottate a diversi operatori o destinazioni, ossia si demultiplexano le chiamate entranti.

Esempio di configurazione per il Per-Call CLI Handling

Quella che segue è la documentazione necessaria ad effettuare il setup del CLI dinamico su Asterisk PBX: va da sé che tali configurazioni sono del tutto adattabili ad altri software PBX ad esso simili, quali, ad esempio 'FreeSwitch' o 'FreePBX'.

1) Di seguito riporto un estratto standard del "sip.conf" di un Asterisk con un'utenza Messagenet di esempio:

```
[sip.messagenet.it]
type = peer
auth = md5
username = DisplayName ; AKA 'defaultuser'
fromuser = 5xxxxxx
secret = password
host = sip.messagenet.it
port = 5061
qualify = yes
insecure = invite,port
```

N.B.: ovviamente sono stati anonimizzati i dati di account e password: per quanto riguarda eventuali differenti impostazioni, non dovrebbero esserci problemi in relazione al fine in discussione.

Fino a questo punto la specifica del CLI in uscita non entra in gioco: quanto visto è necessario puramente al funzionamento dell'utenza sul PBX.

2) Impostazione del CLI dinamicamente sulla base della chiamata:

Passiamo ora ad analizzare come ottenere il CLI dinamico: esistono due modalità distinte per ottenere questo scopo, basate su due diversi Header SIP.

Nel caso si sia deciso di utilizzare una qualsiasi delle due funzionalità descritte, sarà necessario comunicarne preventivamente a Messagenet l'intenzione, al fine di:

- consentire alla vostra utenza di utilizzare quanto scelto;
- configurare la vostra utenza coi numeri telefonici PSTN corretti (da voi comunicati tra quelli assegnati);

2.A) scelta CLI tramite Header "P-Preferred-Identity"

Questa modalità è quella preferibile ed è configurabile unicamente tramite Dialplan del software PBX in questione.

Ipotizziamo di aver abilitato il numero 021234567 e lo 061234567 sulla medesima utenza 5xxxxxx; per scegliere il CLI nelle chiamate in uscita in questa modalità, andrà aggiunto l'header SIP atto allo scopo:

Se ad un certo momento del Dialplan, alla priorità "n", l'estensione "ext" effettua una Dial() SIP usando l'account 5xxxxxx per chiamare 3212345678, normalmente avremmo:

```
...
exten => ext,n,Dial(SIP/5xxxxxx/3212345678)
...
exten => h,1,HangUp()
```

Ora, noi vogliamo scegliere con quale dei due numeri uscire, e per questo fine dobbiamo aggiungere con una priorità precedente alla Dial() un'altra direttiva (le priorità sono tipicamente numeriche, io scrivo "n-1" e "n-2" per farmi capire) atto a scegliere con quale CLI uscire su rete telefonica:

```
...
exten => ext,n-1,SipAddHeader(P-Preferred-Identity: <sip:+39021234567@sip.messagenet.it>)
exten => ext,n,Dial(SIP/5xxxxxx/3212345678)
...
exten => h,1,HangUp()
```

Con questo setup usciamo su rete telefonica tradizionale con CLI 021234567 e la visibilità o meno dello stesso ("presentazione", in gergo) è demandata alle impostazioni dell'utenza Messagenet stessa.

2.B) scelta CLI tramite 'Display-Name' dell'Header "From:"

Come anticipato, e' a disposizione un'ulteriore modalita' per la scelta dinamica del CLI la quale, anziche' basarsi sull'header "P-Preferred-Identity" si basa sul 'Display-Name' proprio dell'header "From:". Anche qui, ipotizziamo di aver abilitato il numero 021234567 e lo 061234567 sulla medesima utenza 5xxxxxx;

Differentemente dal caso 2.A, possiamo procedere in due modalita' distinte:

- la prima e' dinamica e legata, come la precedente, al dialplan del software PBX:

```
...
exten => ext,n-1,Set(CALLERID(name)="+39061234567")
exten => ext,n,Dial(SIP/5xxxxxx/3212345678)
...
exten => h,1,HangUp()
```

- la seconda non e' dinamica e si basa invece sulle configurazioni dell'account SIP; riprendendo quanto precedentemente esemplificato nel punto 1:

```
[sip.messagenet.it]
type = peer
auth = md5
username = "+39061234567" ;; AKA 'defaultuser'
fromuser = 5xxxxxx
secret = password
host = sip.messagenet.it
port = 5061
qualify = yes
insecure = invite,port
```

In entrambi i casi visti, con questo setup usciamo su rete telefonica tradizionale con CLI 061234567 e la visibilita' o meno dello stesso ("presentazione", in gergo) e' demandata alle impostazioni dell'utenza Messagenet stessa.

Per riepilogo, quello che ci aspetteremmo in questo caso dovrebbe essere:

```
From: "+39061234567" <sip:5xxxxxx@sip.messagenet.it>;tag=D0UvUD3S8cFFr
```

3) Impostazione della visibilita' o meno del CLI su linea telefonica tradizionale, Differentemente da quanto riguarda meramente SIP, la "presentation" del numero su PSTN dovra' essere effettuata scegliendo tra due differenti modalita':

- Staticamente e per tutte le chiamate, chiedendoci di impostare la visibilita' o meno del numero telefonico sui nostri sistemi, oppure tramite sito Web (se disponibile);
- Dinamicamente, dopo aver garantito in generale la visibilita' del numero nella modalita' appena definita "statica", aggiungendo poi anche l'Header SIP "Privacy" tramite Dialplan, similmente a quanto gia' precedentemente visto:

```
...
exten => ext,n-2,SipAddHeader(P-Preferred-Identity: <sip:+39021234567@sip.messagenet.it>)
exten => ext,n-1,SipAddHeader(Privacy: none)
exten => ext,n,Dial(SIP/5xxxxxx/3212345678)
...
exten => h,1,HangUp()
```

Con questo setup usciamo su rete telefonica tradizionale con CLI 021234567 e la visibilita' del numero sara' consentita.

Per chiarire, indico che l'Header "Privacy" puo' essere valorizzato con due valori:

- 'none' -> consente la visualizzazione del CLI;
- 'id' -> nasconde il CLI su rete telefonica tradizionale;

N.B.: il fatto di aver utilizzato la modalita' 2.A per la scelta del CLI in uscita non vieta di utilizzare una differente combinazione tra i setup precedentemente visti

4) Appendice e chiarimenti (ATTENZIONE!),

- Il numero telefonico scelto dinamicamente dovra' essere regolarmente a voi assegnato ed e' necessario che venga a noi data comunicazione della volonta' di utilizzare tale numero e quale delle due modalita' (2.A oppure 2.B) si avra' intenzione di usare;

- Il vostro PBX dovra' necessariamente impostare il CLI da usare in formato e164 con leading plus (formato internazionale, preceduto dal segno '+');

- Un punto da chiarire e' relativo all'header "From:" della transazione SIP di INVITE, il quale deve necessariamente riportare l'account SIP in uso (ovvero 5xxxxxx) nella sua 'User-Part'.

Questo a discapito di eventuali fraintendimenti, poiche' capita che si sia tentati di considerare proprio la "User-Part" dell'header "From:" quale punto ove inserire il numero telefonico chiamante, cosa non vera per gli account Messagenet. E' possibile selezionare dinamicamente il CLI tramite la modalita' sopra nominata 2.B, ma si sta parlando del 'DisplayName', non della 'User-Part' della "From" URI;

- Per quanto riguarda la possibilita' di nascondere o meno la presentazione del callerID al chiamato, si rimanda a quanto gia' visto per l'Header Privacy (maniera piu' corretta) oppure rimane per voi possibile, anche se sconsigliato, la possibilita' di specificare nelle modalita' di cui sopra un callerID fittizio e non a voi assegnato in modo tale che nessun

numero sia visualizzato, posto di non aver impostato un callerID di default e che in mancanza dello stesso la presentazione del numero sia ristretta;

- Per concludere, preme precisare che le due chiamate a SipAddHeader() non devono necessariamente essere consecutive e che a voi e' ovviamente demandata la logica di selezione a dialplan tra un CLI e l'altro e la relativa "privacy";

Per qualsiasi necessita', il supporto e' a disposizione per ogni ulteriore chiarimento.

Debugging ricezione DTMF

Facile: editare `/etc/asterisk/logger.conf` e assicurarsi che sia abilitato il logging dei DTMF. A me piace metterli a parte, aggiungendo la riga

```
dtmf => dtmf
```

alla sezione `[logfiles]`. Questo produrrà un file di nome `dtmf` nella cartella dei log di asterisk (di default `/var/log/asterisk`).

Il debugging consiste nel verificare autonomamente, da parte del cliente, che in tutti i casi vengano sempre ricevuti. Se poi permangono dubbi, bisogna fare la stessa cosa con una cattura dell'RTP, per verificare se noi inviamo sempre i DTMF secondo [RFC 2833](#). Questa cattura può essere effettuata anche in autonomia dal cliente sul suo centralino con `tcpdump`, usando ad esempio il comando:

```
tcpdump -qp -i <interfaccia_pubblica> -s0 -w <file_di_cattura> udp and \(( net 212.97.59.64/27 or net 193.227.104.0/24 \)
```

Il file di cattura prodotto poi potrà essere analizzato con comodo.

Procedure per Problemi di NAT Traversal

All'interno del `sip.conf` inserire nella sezione `[general]` le seguenti:

```
registertimeout=60
```

```
nat=yes
```

```
localnet=192.168.0.0/24 ;<-- Sostituire con propria rete locale in notazione CIDR
```