Appunti

Per interagire con il centralino tramite API bisogna interagire con Asterisk che è un modulo disponibile in PBX. Ci sono due modalità di comunicazione con Asterisk:

- Da gestionale verso asterisk: si utilizza ARI (Asterisk Rest Interface)
- Da Asterisk verso gestionale bisogna appoggiarsi ad AMI (Asterisk Manager Interface). Abbiamo sviluppato uno script python che si collega ad AMI, si connette all'evento dello squillo del telefono e al momento dello squillo invia tramite post i dati della telefonata al gestionale.

Configurazioni ARI:

- Abilitare l' IP che può fare le chiamate API

da SSH lanciare il comando: "**nano /etc/asterisk/ari.conf**" per aprire e modificare il file: ari.config

1. Verifica il contenuto di ari.conf

bash

```
cat /etc/asterisk/ari.conf
```

Deve contenere:

ini

```
[general]
enabled = yes
pretty = yes
allowed_origins = *

[admin]
type = user
read_only = no
password = 12345678
```

---

2. Se il file è diverso, modificalo e ricarica, da SSH lanciare i comandi i modo separato:

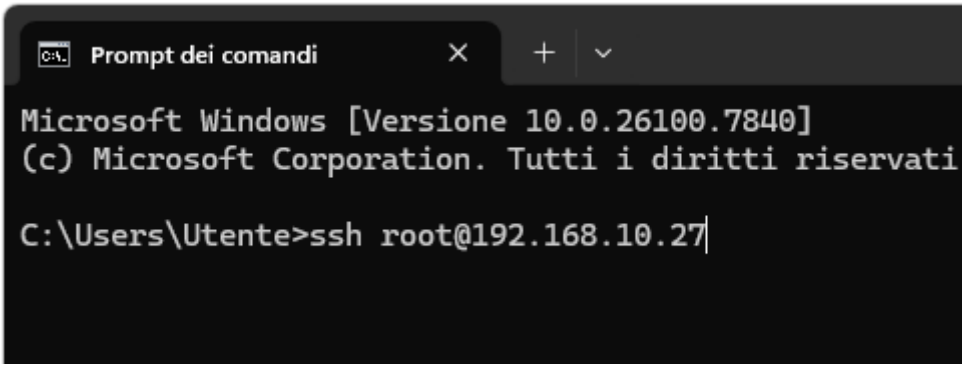
- nano /etc/asterisk/ari.conf
- asterisk -rx "module reload res_ari.so"

3. Per fare una chiamata ARI c'è bisogno dell'utente abilitato:

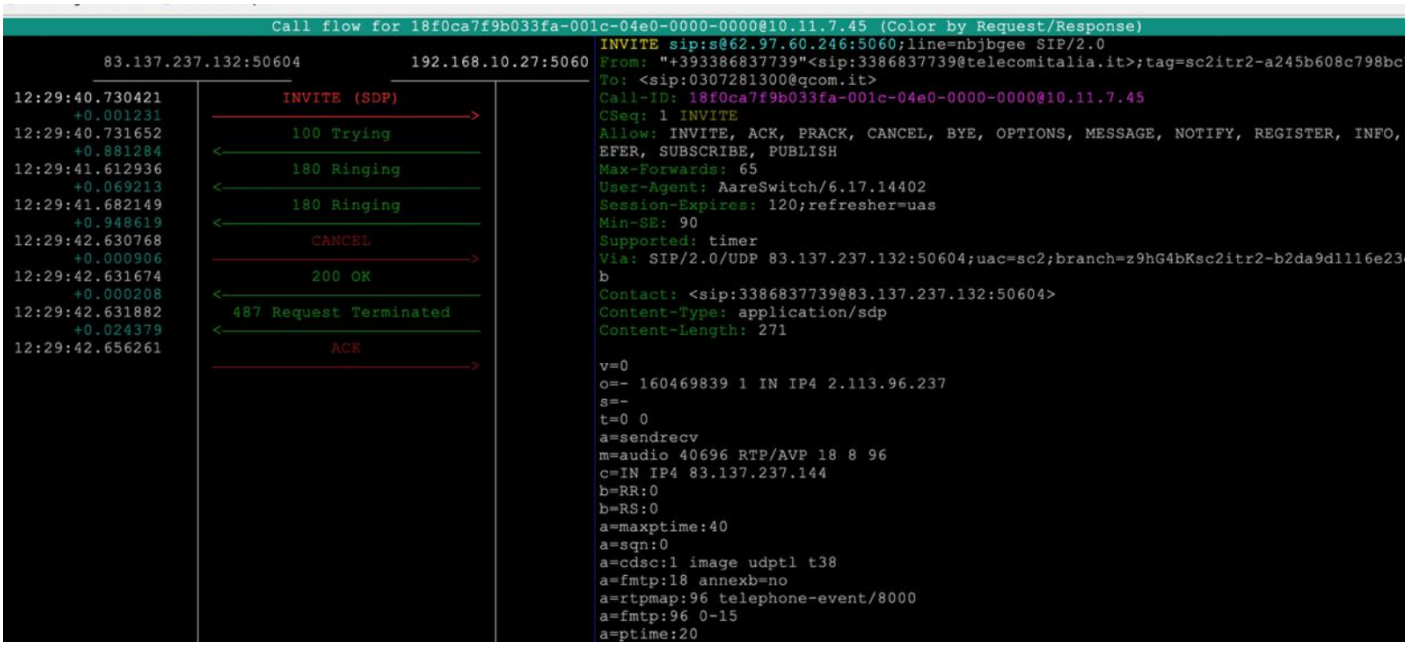
- Modulo "Utenti Asterisk Rest Interface"

CONFIGURAZIONE AMI:

Connessione ssh a PBX



Esempio dati al momento della ricezione della chiamata:

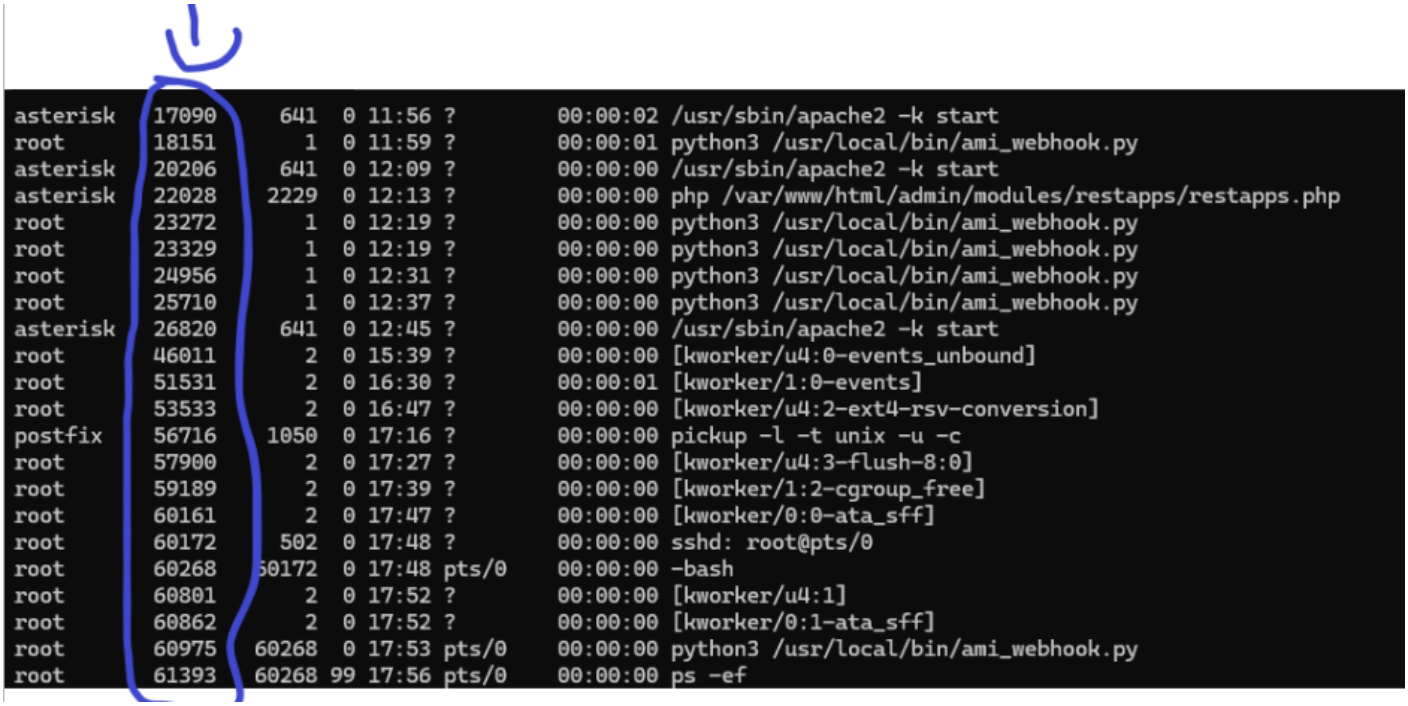


File Phyton in Background in ascolto di AMI (Asterisk Manager Interface):
nano /usr/local/bin/ami_webhook.py

Per Attivarlo in BackGround:
nohup /usr/local/bin/ami_webhook.py &

Per svuotare i servizi in background bisogna KILLARE i processi in esecuzione
ps -ef (serve per vedere tutto l'elenco dei processi)
ps -ef | grep webhook (serve per vedere tutto l'elenco dei processi filtrati x nome file *webhook* .*)

kill -9 PID (dove `PID` è l'ID del processo ottenuto dai comandi precedenti.



```
asterisk   17090     641  0 11:56 ?        00:00:02 /usr/sbin/apache2 -k start
root       18151       1  0 11:59 ?        00:00:01 python3 /usr/local/bin/ami_webhook.py
asterisk   20206     641  0 12:09 ?        00:00:00 /usr/sbin/apache2 -k start
asterisk   22028    2229  0 12:13 ?        00:00:00 php /var/www/html/admin/modules/restapps/restapps.php
root       23272       1  0 12:19 ?        00:00:00 python3 /usr/local/bin/ami_webhook.py
root       23329       1  0 12:19 ?        00:00:00 python3 /usr/local/bin/ami_webhook.py
root       24956       1  0 12:31 ?        00:00:00 python3 /usr/local/bin/ami_webhook.py
root       25710       1  0 12:37 ?        00:00:00 python3 /usr/local/bin/ami_webhook.py
asterisk   26820     641  0 12:45 ?        00:00:00 /usr/sbin/apache2 -k start
root       46011       2  0 15:39 ?        00:00:00 [kworker/u4:0-events_unbound]
root       51531       2  0 16:30 ?        00:00:01 [kworker/1:0-events]
root       53533       2  0 16:47 ?        00:00:00 [kworker/u4:2-ext4-rsv-conversion]
postfix    56716    1050  0 17:16 ?        00:00:00 pickup -l -t unix -u -c
root       57900       2  0 17:27 ?        00:00:00 [kworker/u4:3-flush-8:0]
root       59189       2  0 17:39 ?        00:00:00 [kworker/1:2-cgroup_free]
root       60161       2  0 17:47 ?        00:00:00 [kworker/0:0-ata_sff]
root       60172     502  0 17:48 ?        00:00:00 sshd: root@pts/0
root       60268   60172  0 17:48 pts/0    00:00:00 -bash
root       60801       2  0 17:52 ?        00:00:00 [kworker/u4:1]
root       60862       2  0 17:52 ?        00:00:00 [kworker/0:1-ata_sff]
root       60975   60268  0 17:53 pts/0    00:00:00 python3 /usr/local/bin/ami_webhook.py
root       61393   60268 99 17:56 pts/0    00:00:00 ps -ef
```

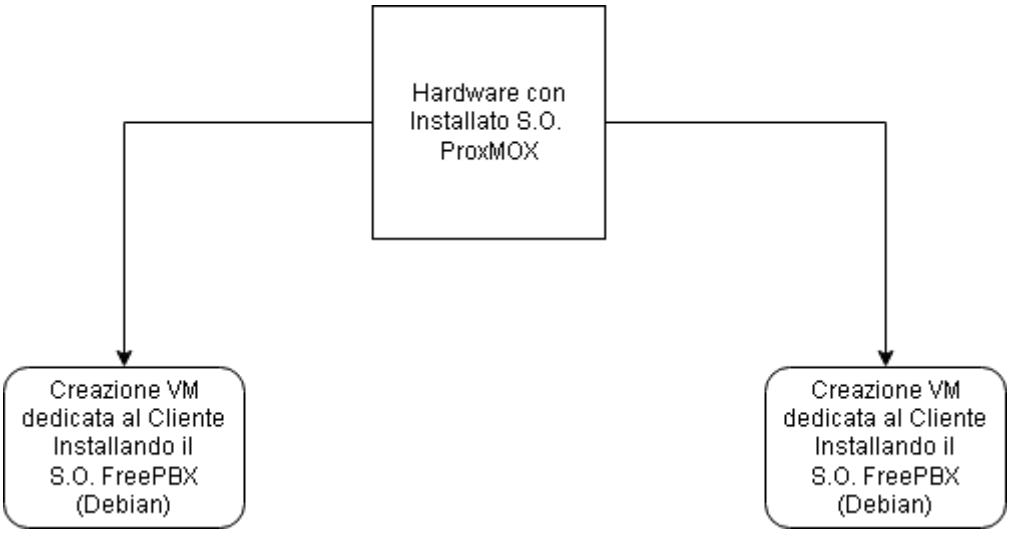
FILE DI CONFIGURAZIONE AMI

sudo nano /etc/asterisk/manager.conf

```
[general]
enabled = yes
port = 5038
bindaddr = 0.0.0.0 ==> aperto a tutti (SOLO PER PROVE)
displayconnects=no ;only effects 1.6+
```

```
[bNW7t3zJv2kO] ==> NOME UTENTE DA METTERE NELLO SCRIPT PYTHON
secret = 08rcxfNjS8hI ==> PASSWORD DA METTERE NELLO SCRIPT PYTHON
deny = 0.0.0.0/0.0.0.0 ==> aperto a tutti (solo per prove)
permit = 0.0.0.0/0.0.0.0 ==> aperto a tutti (solo per prove)
read =
system,call,log,verbose,command,agent,user,config,command,dtmf,reporting,cdr,dialplan,originate,mes
sage
write =
system,call,log,verbose,command,agent,user,config,command,dtmf,reporting,cdr,dialplan,originate,mes
```

```
sage
writetimeout = 5000
```



Revisione #13
Creato 2026-03-03 07:56:04 UTC da Editor
Aggiornato 2026-03-03 17:01:33 UTC da Editor