

Practical Guide



How to setup
an IP04
Asterisk IP-PBX

Table of Contents

1. Background.....	1
2. The VoIP scenarios.....	2
3. Before getting started.....	3
3.1 Scenario 1.....	3
3.2 Scenario 2.....	3
3.3 Scenario 3.....	4
4. Behind the scenes.....	5

1. Background

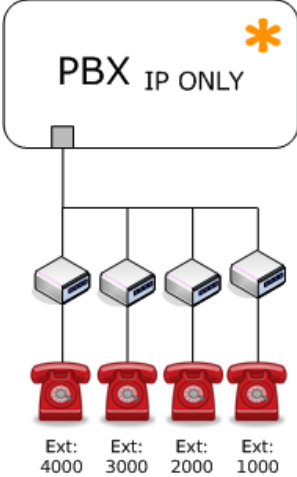
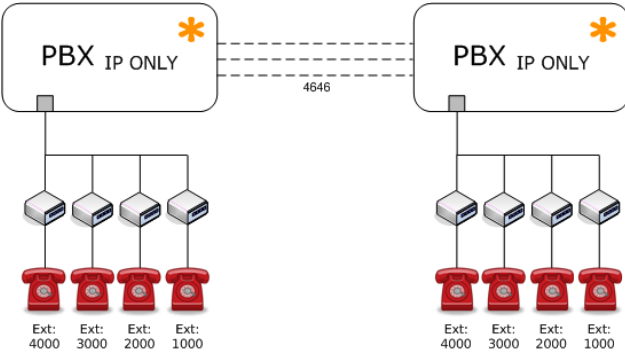
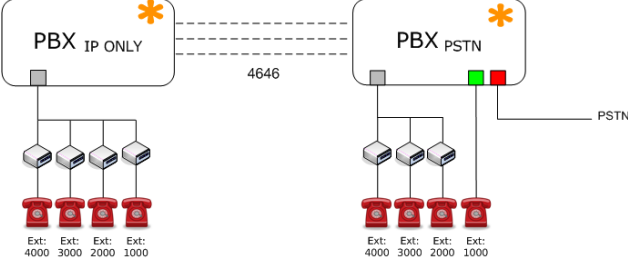
The first edition of the “VoIP-4D Primer, Building voice infrastructure in developing regions” released in December 2006 covered basic IP Telephony and showed how to configure Asterisk for three basic scenarios.

This document shows you how to set up 3 basic scenarios on the IP04 IP-PBX. The IP04 is a tiny, full function Asterisk IP-PBX with 4 analogue ports. It retails for around \$450 but can be potentially be built and deployed to developing regions for under \$100.

Configuration of the IP04 can be performed using just an analogue telephone - even a PC is optional! These exercises show just how easy Asterisk configuration can be on the IP04. Compared to commodity hardware (such a PC and PCI card), IP04 configuration is much easier.

2. The VoIP scenarios

The three scenarios described in this document are similar to the ones described in the *VoIP-4D Primer*.

<p>Scenario 1</p> <p><i>Creating a local private telephony network in a rural community</i></p> <p>This scenario consists of a single IP04 with 1 to 4 analog phones.</p>	
<p>Scenario 2</p> <p><i>Interconnecting communities</i></p> <p>In this scenario we interconnect two IP04 PBXs. Local extensions of one PBX are made available to the other and vice versa.</p>	
<p>Scenario 3</p> <p><i>Connecting communities to the PSTN</i></p> <p>In this final scenario, we have interconnected two PBXs and allow the possibility of reaching the PSTN from any of them.</p>	

3. Before getting started

If you have not read the *VoIP-4D Primer* (www.voip4d.org) please read the first chapter as it will provide you with the necessary background to understand the basic concepts of IP Telephony.

You will require two IP04s. Both IP04s should be configured with two FXO (Port1 and Port2) and 2 FXS ports (Port3 and Port4). Both IP04s should be configured using the scripts contained in:

http://www.rowetel.com/ucasterisk/downloads/voip4d_lab_ip04_001.tar.gz

See the README in the tar ball for configuration information.

The two IP04s can be connected via your LAN, a WiFi connection, or even an Ethernet crossover cable.

This documentation has used IP addresses in the range 192.168.46.0/24. Be aware that you may need to use your own IP address range, for example if operating on a LAN or over a WiFi link.

3.1 Scenario 1

1. Apply power to the IP04. After about 60 seconds it will be ready for use.
2. Plug phones into Port3 and Port4. Check that you hear dial tone in each phone.
3. Dial 3000 to reach Port3, or dial 4000 to reach Port4.

3.2 Scenario 2

In this scenario, we will place a phone call from a FXS phone on *IP04-A* to a FXS phone on *IP04-B*. *IP04-A* will connect to *IP04-B* using the SIP protocol over an IP link.

We will use the prefix 9 to reach extensions on the other PBX. So dialling *9+4000* on a phone connected to *IP04 A* will place a call to Port4 on IP04 B.

We can still reach extension on our own PBX by simply dialling the extension number.

Each IP04 will be configured as a SIP “peer” of the other IP04. We will set up the IP of each IP04, and the IP of it's SIP peer:

IP04	IP	IP of SIP Peer
IP04-A	192.168.46.31	192.168.46.32
IP04-B	192.168.46.32	192.168.46.31

Configuration of IP04-A

1. Hardware setup
 - Power up IP04-A and wait for it to boot up.
 - Plug a phone into Port4.
2. Configure IP address
 - Dial “C-O-N-F” (2663), then “dial” the IP of IP04-A (192*168*46*31#).
The IP will be read back to you.
3. Configure IP address of SIP peer
 - Dial “S-I-P” (747), then “dial” the IP of the SIP peer (192*168*46*32#).
The IP of the SIP peer will be read back to you.

Configuration of IP04-B

1. Hardware setup
 - Power up IP04 B.
 - Plug a phone into Port4
2. Configure IP address
 - Dial “C-O-N-F” (2663), then “dial” the IP of IP04-B (192*168*46*32#).
The IP will be read back to you.
3. Configure IP address of SIP peer
 - Dial “S-I-P” (747), then “dial” the IP of the SIP peer (192*168*46*31#).
The IP of the SIP peer will be read back to you.

Now, try to dial between the two IP04-PBXs by using the prefix 9:

1. Dial 9+4000 on IP04-A to reach Port4 on IP04-B
2. Dial 9+4000 on IP04-B to reach Port4 on IP04-A.

3.3 Scenario 3

This scenario is very similar to 2, except that we will place a call from a FXS port on IP04-A to a FXO port on IP04-B. In this case the IP04-A to IP04-B link is being used to “extend” the PSTN over an IP link.

1. Configure IP04-A and IP04-B as in Scenario 2.
2. Connect an FXO line (PSTN) to Port1 of IP04-B.
3. Pick up the phone connected to Port4 of IP04-A. Dial 90 on this phone to connect to the FXO line on Port1 of IP04-B.

4. Behind the scenes

In this section the configuration of the IP04 is examined in more detail. This section requires a PC with telnet to connect to the IP04.

On you host PC “telnet 192.168.46.31”, this will produce a root prompt on the IP04. By default no user name or login is required:

```
[host]$ telnet 192.168.46.31
Trying 192.168.46.31...
Connected to 192.168.46.31.

Escape character is '^]'.
BusyBox v1.4.1 (2007-10-09 10:23:35 CST) Built-in shell (msh)
Enter 'help' for a list of built-in commands.

root:~>
```

You can check the IP04 IP after dialling “C-O-N-F” (2663) using ifconfig:

```
root:~> ifconfig
eth0      Link encap:Ethernet  HWaddr 00:09:45:54:20:B1
          inet addr:192.168.46.31 Bcast:192.168.46.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4737 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4892 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          Interrupt:48
```

To obtain an Asterisk command line (CLI) interface type “asterisk -r”:

```
root:~> asterisk -r
afl*CLI>
```

You can check if IP04-A and IP04-B are set up as SIP peers using the “show sip peers” command”:

```
afl*CLI> sip show peers
Name/username      Host                Dyn Nat ACL Port      Status
ast/ast            192.168.46.32      0   0   0   5060     OK (2 ms)
```

You can examine the section of the Asterisk dial-plan that handles IP configuration:

```
afl*CLI> show dialplan 2663@default
[ Context 'default' created by 'pbx_config' ]
'2663' => 1. Read(newip||15) [pbx_config]
          2. SayDigits(${newip}) [pbx_config]
          3. System(/bin/echo "${newip}" > /tmp/newip.txt) [pbx_config]
          4. System(/bin/setnewip) [pbx_config]
```

This dial-plan fragment calls the script `/bin/setnewip`. If you exit the Asterisk CLI, you can examine this shell script:

```
root:/etc/asterisk> cat /bin/setnewip
#!/bin/sh -x
# /bin/setnewip
# David Rowe Nov 18 2007
#
# Sets a new IP based on DTMF digits collected by Asterisk
# Asterisk places it's collected IP in /tmp/newip.txt

if [ ! -f /tmp/newip.txt ]; then
    exit
fi

# convert * separated IP like 192*168*46*32 to 192.168.46.32

cat /tmp/newip.txt | sed 's/\*/\./g' > /tmp/newip_dot.txt

# Modify /etc/asterisk/rc.conf static IP, so we reboot with
# the correct IP

NEWIP=`cat /tmp/newip_dot.txt`
cat /etc/asterisk/rc.conf | sed
"s/IPADDRESS=\".*\..\..\..\..\"/IPADDRESS=\"${NEWIP}\"/" > /tmp/rc.conf
cp /tmp/rc.conf /etc/asterisk/rc.conf

# ifconfig to set current IP

ifconfig eth0 $NEWIP
```