VanNet Setup & Operations Guide

Cupertino Amateur Radio Emergency Service

September 2014

Revision 0.3

DRAFT

Table of Contents

Revision

Revision	Date	Comments
0.1	8/27/2014	Initial Draft
0.2	9/01/2014	Draft, Update

1 Introduction

1.1 Purpose

This document describes the Cupertino OES Communications Van Network, how to set it up, its operation, and maintenance.

1.2 Distribution

This plan is to be distributed to all Cupertino Amateur Radio Emergency Service (CARES) Van Supervisors and others with responsibility for setting up the Communications Van. CARES will use this document during deployments as well as for training.

1.3 Effective Date

This document is effective October 1, 2014. Updates to this document will be distributed as required by CARES.

1.4 How to use this document

This document includes information that is useful to CARES Van Supervisors.

- New and existing Van Supervisors should use this document to understand the VanNet system and its operation.
- The CARES leadership will use this document as a reference point to making improvements to the VanNet system.

1.5 Revisions

CARES will review this document on an as-needed basis. Changes to this document (either individual pages or in its entirety) will be distributed to all Van Supervisors and Cupertino OES staff.

1.6 References

None listed.

2 System Description

2.1 Introduction

The Comm Van network is a local area network independent of the Internet designed for the sole purpose of moving information between the Comm Van and the DOC.

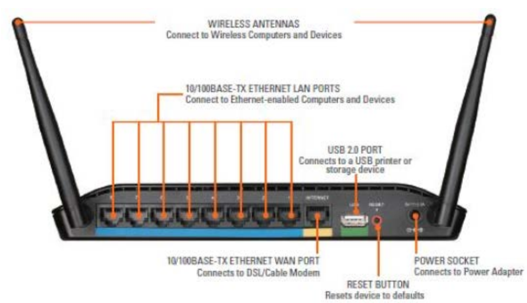
The specific capabilities that the network will support include:

1. VoIP phone system. A local area phone system will be implemented for direct communication between the Comm Van and the DOC.
2. File Share. Data files can be placed on a file share that is commonly accessible by the Comm Van and the DOC.
3. Chat. This capability allows the two locations to exchange quick text messages when a phone call is not necessary.

2.2 System Overview

The system is a mix of both fixed and deployable components, and will be set up in 1of 2 locations. Refer to the diagram on the following page for a description of how all the pieces fit together.

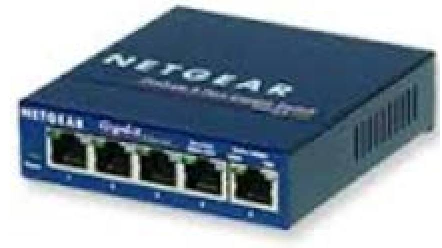
Communications Van Side

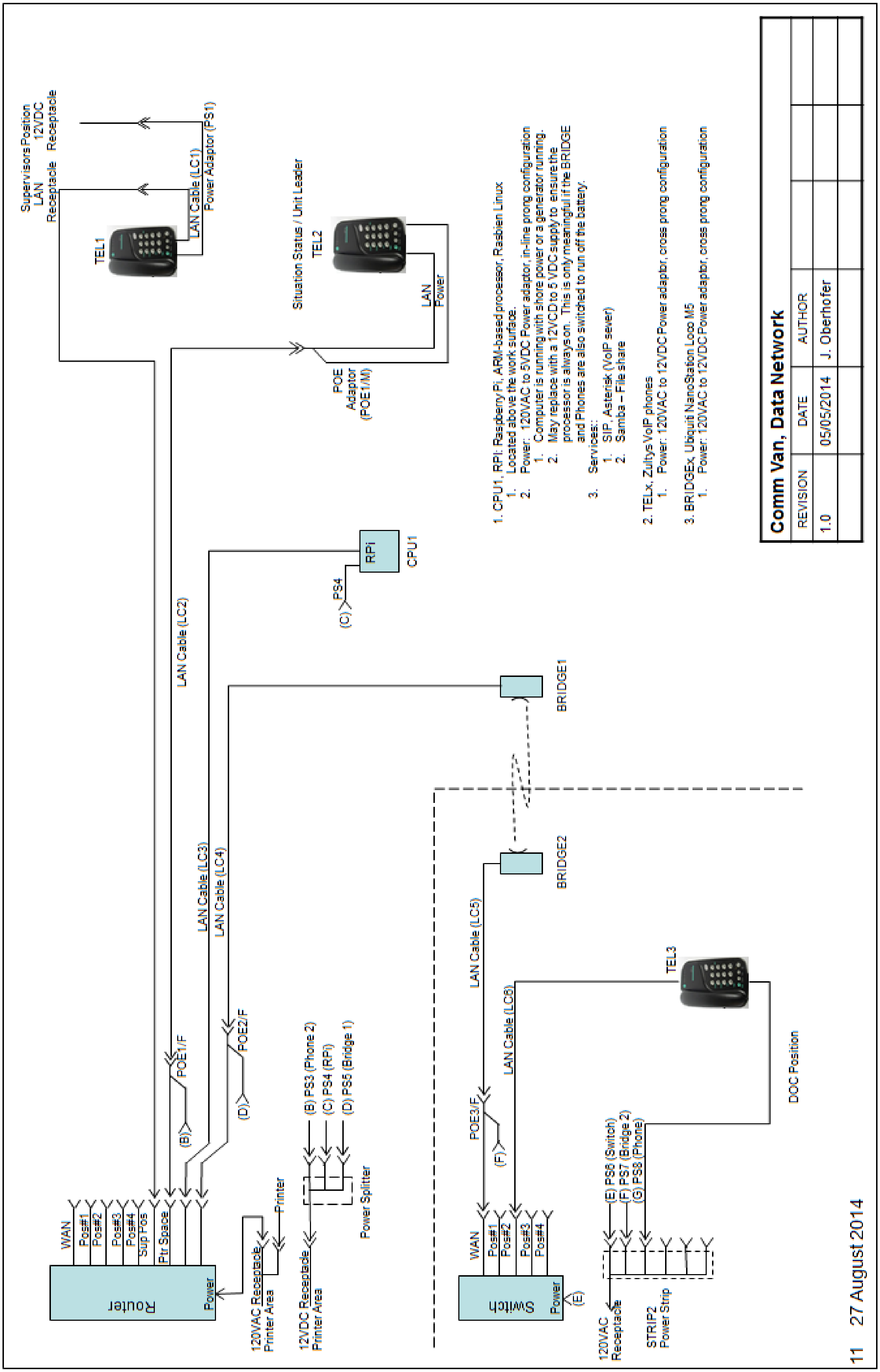
<table>
<tr><td>Component</td><td>Designator</td><td>Description</td></tr>
<tr><td>Server</td><td>CPU1</td><td>Linux server that hosts all the network services used to support the VanNet network. The server is permanently mounted to the inside of the Van on the wall between Operator positions #1 and #2.
POWER: 12VDC
</td></tr>
<tr><td>Router</td><td>Router</td><td>D-Link DIR-632 Router
Network distribution for all devices located in the vicinity of the Comm Van. This is permanently installed in the Van in the printer drawer.
POWER: 120VAC power adaptor.

</td></tr>
</table>

Phone	TEL1	**Zultys ZIP2 VoIP Phone** Supervisor's position. This is a fixed installation. POWER: 12VDC 
Phone	TEL2	**Zultys ZIP2 VoIP Phone** SitStat position. Same phone as above. This is a deployable phone that is to be set up somewhere outside of the van, and hardwired into the van' network This phone requires a **POE-Adaptor** (Power over Ethernet) on both ends of the CAT5 cable. POWER: 12VDC 
Wi-Fi Bridge	BRIDGE1	**Ubiquiti NanoStation Loco M5 (5Ghz)** This is the van side of the Wi-Fi link with the DOC. The device is stowed inside the van and deployed whenever communications is required with the DOC. Uses Power over Ethernet (POE) power module located in the van. POWER: 120VAC 

DOC Side

Wi-Fi Bridge	BRIDGE2	**Ubiquiti NanoStation Loco M5 (5Ghz)** Same as the VAN-side Wi-Fi Bridge. This is the EOC side of the Wi-Fi link to the Van. This device is stowed in the vicinity of where the DOC will set up, and deployed whenever communications is required with the Van. Both sides must be set up for the two-way communications to occur. POWER: 120VDC, Power over Ethernet (POE)

Network Switch	Switch	**NetGear 5 Port Switch** Network distribution for all devices located in the vicinity of the DOC. This device is set up at the time of deployment with the rest of the DOC side of the solution. POWER: 120VAC power adaptor 
Phone	TEL3	**Zultys ZIP2 VoIP Phone** DOC position. This is a deployable phone that is set up at the time of deployment with the rest of the DOC side of the solution. POWER: 120VAC power adaptor (Not a POE-Adaptor for this phone)



Comm Van, Data Network
REVISION DATE AUTHOR
1.0 05/05/2014 J. Oberhofer
1. CPU1, RPi: Raspberry Pi, ARM-based processor, Rasbien Linux
1. Located above the work surface.
2. Power: 120VAC to 5VDC Power adaptor, in-line prong configuration
1. Computer is running with shore power or a generator running.
2. May replace with a 12VCD to 5 VDC supply to ensure the processor is always on. This is only meaningful if the BRIDGE and Phones are also switched to run off the battery.
3. Services:
1. SIP, Asterisk (VoIP sever)
2. Samba – File share
2. TELx, Zultys VoIP phones
1. Power: 120VAC to 12VDC Power adaptor, cross prong configuration
3. BRIDGEx, Ubiquiti NanoStation Loco M5
1. Power: 120VAC to 12VDC Power adaptor, cross prong configuration
Supervisors Position
LAN Receptacle
12VDC Receptacle
TEL1
LAN Cable (LC1)
Power Adaptor (PS1)
Situation Status / Unit Leader
TEL2
LAN
Power
POE Adaptor (POE1/M)
RPi
CPU1
PS4
(C)
LAN Cable (LC2)
BRIDGE1
BRIDGE2
LAN Cable (LC3)
LAN Cable (LC4)
LAN Cable (LC5)
LAN Cable (LC6)
TEL3
DOC Position
POE1/F
POE2/F
POE3/F
(B)
(D)
(F)
(B) PS3 (Phone 2)
(C) PS4 (RPi)
(D) PS5 (Bridge 1)
Power Splitter
Printer
WAN
Pos#1
Pos#2
Pos#3
Pos#4
Sup Pos
Ptr Space
Router
Power
120VAC Receptacle Printer Area
12VDC Receptacle Printer Area
Switch
(E)
(E) PS6 (Switch)
(F) PS7 (Bridge 2)
(G) PS8 (Phone)
120VAC Receptacle
STRIP2 Power Strip
11 27 August 2014

3 Comm Van Network Setup

This section describes how to deploy and turn on the network components.

3.1 Network Router

Information	The router is fixed mounted inside the Van's printer cabinet. Once power is applied, the router will boot and up and register all connected network devices. Because some devices may need to be power-cycled for their IP addresses to be issued, it is recommended to power up this component first.
Set up	Nothing to do
Power up	120VAC. Start the generator or ensure shore power is connected. 120VAC can be confirmed by looking at the AutoCharge Panel LEDs at the rear wall of the van. All 120VAC CBs should be set to ON. No additional user interaction is required to apply power.
Verify	Observe the LEDs on the Router show which router ports are connected.

3.2 Linux Server

Information	The server is permanently mounted to the inside of the Van on the wall between Operator positions #1 and #2.
Set up	Nothing to do. Once power is applied, the server will boot up and Linux services will become available.
Power up	12VDC Distribution Panel, turn on 12VDC CB5. No additional user interaction is required to apply power.
Verify	Observe the LEDs on the Server show server activity.

3.3 VoIP Phone, Supervisor's Position

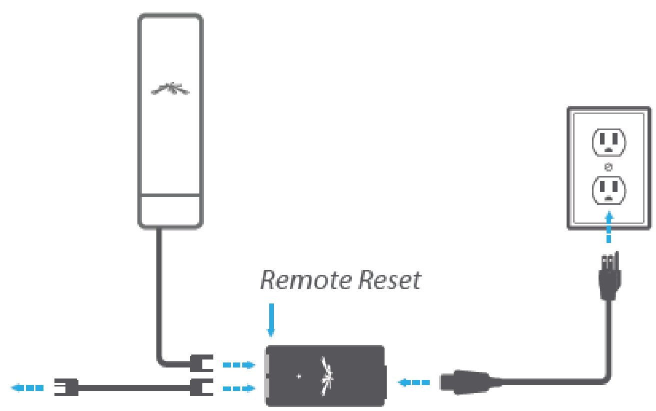
Information	This telephone is located the supervisor's position.
Set up	Nothing to do. The phone is permanently installed and cabled to the network and power connectors at the supervisor's station.
Power up	Make sure the router is running first. 12VDC Distribution Panel, turn on 12VDC CB5. No additional user interaction is required to apply power.
Verify	1. Phone boots up, light flashes twice. 2. Phone registers with the Router to get its IP address. Once the IP address is assigned, a Dial Tone can be heard. 3. Phone then registers with the Asterisk Server (running on the Linux Server). Once registration is complete, you can then dial the phone.

	4. *Test:* Dial extension 500 for the loop-back test to confirm the phone is talking to the Asterisk sever.

3.4 VoIP Phone, Exterior Position

Information	The phone can be deployed anywhere within the vicinity of the Van.
Set up	1. Retrieve the Network box from the van. 2. Unpack the phone, and place it where it is to be used. 3. Verify the POE Adaptor to the phone is connected (both the power connector and the Ethernet connector). 4. The CAT5 cable for this phone is located below Operating Position #1 in the van. Uncoil the cable and connect the loose end to the POE Adaptor. ***NOTE:*** If the phone is deployed outside the Van, ensure that the CAT5 cable is clear of doors or other areas that may pinch or cut the cable. 
Power up	Make sure the router is running first. 12VDC Distribution Panel, turn on 12VDC CB5. No additional user interaction is required to apply power.
Verify	5. Phone boots up, light flashes twice. 6. Phone registers with the Router to get its IP address. Once the IP address is assigned, a Dial Tone can be heard. 7. Phone then registers with the Asterisk Server (running on the Linux Server). Once registration is complete, you can then dial the phone. 8. *Test:* Dial extension 500 for the loop-back test to confirm the phone is talking to the Asterisk sever.

3.5 Wi-Fi Bridge

Information	The Ubiquiti NanoStation must be deployed whenever connectivity is needed between the DOC (or wherever the remote Network components are set up) and the Van. If remote internet access is not needed, there is no need to deploy this device.
Set up	The goal is to mount the NanoStation so that is it pointing to the client portion of the Wi-Fi Bridge.  1. Unpack the Ubiquiti NanoStation and CAT5 cable. The CAT5 Cable should already be connected to the NanoStation. 2. Mount the NanoStation Loco. For standard Van parking placement at the EOC, • Mount the NanoStation on the upper portion of the left vertical ladder support. • Ensure the Rubber pad is between the NanoStation mount and the ladder support. • Secure the NanoStation to the ladder support using a tie wrap. • Ensure the NanoStation is not loose. 3. For non-standard Van placement at the EOC, find an appropriate location where you can mount the NanoStation similar to what is described above. 4. Inside the Van, plug in the other end of the CAT5 cable into the POE module at Station #1. 
Power up	120VAC. 1. Ensure the generator or Shore Power is applied. 2. Ensure the POE Module is plugged in.
Verify	1. Verify the POE Module Power-on LED illuminates. 2. Verify the LEDS on the back of the NanoStation illuminate.

4 DOC Network Setup

All components described here are located in the VanNet Network Storage box, located above Ken's desk, left side.

4.1 Network Switch

Information	
Set up	1. Remove the power strip, and set it up in a convenient location; plug it into a 120VAC outlet. 2. Remove the Switch from the Network Storage box. 3. Connect the Power Adaptor 12VDC plug to the Switch. Plug in the Power Adaptor into the power strip.
Power up	Requires 120VAC to operate. Other than performing the setup steps above, no additional user interaction is required to apply power.
Verify	Observe the LEDs on the Router show which router ports are connected.

4.2 VoIP Phone, DOC Position

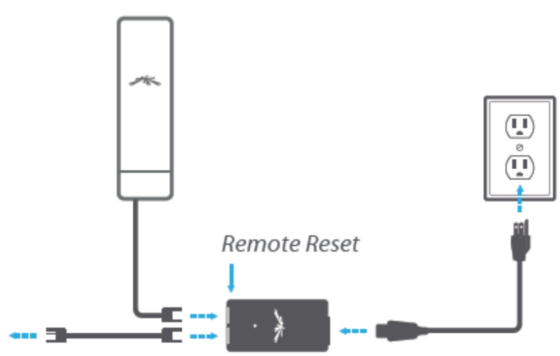
Information	The phone is stored in the DOC's VanNet Network storage box. This device may need to power-cycled for its IP address to be issued. It is recommended the Van side of the Network Bridge be powered up first. If this is not the case, you may need ot power-cycle this phone after the van portion of the network comes up.
Set up	1. Unpack the phone, and place it where it is to be used. 2. Connect the Phone CAT5 cable between the Phone and one of the Switch ports plugs. 3. Connect the Phone Power Adaptor, 12VDC connector to the phone power connector. 4. Connect the Phone Power Adaptor to the power strip.
Power up	Requires 120VAC to operate. See the 120VAC power-up sequence for the Network Switch.
Verify	1. Phone boots up, light flashes twice. 2. Phone registers with the Router to get its IP address. Once the IP address is assigned, a Dial Tone can be heard. 3. Phone then registers with the Asterisk Server (running on the Linux Server). Once registration is complete, you can then dial the phone. 4. *Test:* Dial extension 500 for the loop-back test to confirm the phone is talking to the Asterisk sever.

4.3 Wi-Fi Bridge

The Ubiquiti NanoStation must be deployed whenever connectivity is needed between the DOC (or wherever the remote Network components are set up) and the Van. If remote internet access is not needed, there is no need to deploy this device.

The NanoStation is stored in the DOC's VanNet Network storage box.

<table>
<tr>
<td>Set up</td>
<td>
The goal is to mount the NanoStation so that is it pointing to the access point side of the Wi-Fi Bridge.
1. Make sure you have the following parts:
• Ubiquiti NanoStation Loco M5
• CAT5 cable
• Window Mount
2. Attach the CAT5 cable to the NanoStation.
3. Mount the NanoStation Loco. For standard Van placement at the EOC,
• Assemble the Window Mount per the picture below.
• Attach the NanoStation to the window mount so that the NanoStation antenna is facing in the same direction as the suction cup and the window.
• Attach the NanoStation window mount to the window by placing the suction cup against the window, and then pushing the suction cup lever down.
4. Verify the attachment is secure before walking away.

5. Plug in the NanoStation CAT5 cable to the POE Injector.
6. Plug in the short section of CAT5 cable between the POE Injector and the Router.


</td>
</tr>
</table>

Power up	Requires 120VAC to operate. See the 120VAC power-up sequence for the Network Switch. Plug in the POE injector into an 120VAC outlet.
Verify	1. Verify the POE Module Power-on LED illuminates. 2. Verify the LEDS on the back of the NanoStation illuminate.

5 Device Configurations

The following table describes the setup for each piece of network equipment.

IP Address Table

Router	192.168.0.1
Raspberry Pi	192.168.0.4
NanoStation Loco M5 Access Point	192.168.0.2
NanoStation Loco M5 Client	192.168.0.3
Telephone1	DHCP
Telephone2	DHCP
Telephone3	DHCP

Router

Mfgr. Model		D-Link, DIR-632
Network Setting	MAC Address	WAN= 1c:af:f7:dc:87:18 LAN= 1c:af:f7:dc:87:17 Wireless= 1c:af:f7:dc:87:17
	IP Addresss	192.168.0.1
	Logon	admin
	Password	<blank>
	Access	Browser
Wireless Settings	Wireless LAN SSID	COMMVAN
	8021.11 Mode	11gn
	Channel Width	20
	Security Mode	AUTO (WPA or WPA2) – PSK
	Preshared Key	cvan2012

Linux Server

Mfgr, Model	Raspberry Pi,
MAC Address	b8:37:eb:9e:f8:87
IP Addresss	192.168.0.4
Logon	Pi
Password	Raspberry
Access	ssh <ip addr> -l pi

Telephones, VoIP

Mfgr, Model	Zultys ZIP2			
Tab	Setting	Telephone 3 DOC	Telephone 2 SIT STAT	Telephone 1 VAN
	MAC Address	00:01:e1:09:de:6d	00:01:e1:09:de:23	00:01:E1:09:DE:76
	IP Address	DHCP*	DHCP*	DHCP*
	Logon			
	Password			
	Access	Browser	Browser	Browser
SIP, SIP Server Settings	Server Addr	192.168.0.101	192.168.0.4	192.168.0.4
	Port	5060	5060	5060
	Domain Name	192.168.0.101	192.168.0.4	192.168.0.4
	Send Ref Req	☑	☑	☑
SIP, SIP gateway settings	Dial Plan		x.T|x.#|*x.T|*x.#	x.T|x.#|*x.T|*x.#
	Transport		UDP	UDP
	Phone No		CUPSITL	CUPVAN
	Caller ID		CUPSITL	CUPVAN
	Port	5060	5060	5060
	AEC ON		ON	ON
	User Name	CUPEOC	CUPSITL	CUPVAN
	Pwd	eoc	sitl	van
CODECS	G711U		20ms, OFF	20ms, OFF
	G711A		20ms, OFF	20ms, OFF
	☑ G729		20ms, OFF	20ms, OFF
CODECS, Jitter Buffer	⦿ Adaptive Jitter Buffer		100ms	100ms
	○ Fixed Jitter Buffer		40ms	40ms
	☐ Auto switch…		Unchecked	Unchecked
CODECS, Paging support	☐ Enable		Unchecked	Unchecked
	Paging Server			
	Paging Port		3771	3771
	Paging Codec		G700U	G700U

* Telephones are assigned IP Addresses by DHCP. Because IP addresses may be different from one router power up to another, you will have to connect to the Router and navigate to the SETUP, Network Settings Tabs to compare the MAC addresses with those in the DHCP table. HOSTNAME for these phones is listed as **UNKNOWN**.

Ubiquiti NanoStation Loco M5

Tab	Setting	AP	Client
Wireless Tab	Mode	Access Point	Station
	SSID	Bridge	bridge
	Outpost Power	4	4
	Security	WPA2-AES/PSK	WPA2-AES/PSK
	Key	vanlink2014	vanlink2014
Network Tab	IP	192.168.0.2	192.168.0.3
	Net Mask	255.255.255.0	255.255.255.0
Adv Tab	Enable AirMax	✓	✓
	Distance	0.4 Miles	0.4 Miles
	AirMax Priority		
	Logon	unbt	unbt
	Password	unbt	unbt

6 Troubleshooting

6.1 Telephone Issues

Problem	*Symptom:* Phone lights flash twice, but no Dial Tone.
Solution	*Check:* The router is not on. Start the router per the above, and cycle power on the phone to reboot the phone (unplug, and then plug back in the power line). *Check:* The phone network connection is not in. Ensure the network cable to the phone is connected, and cycle power on the phone to reboot the phone (unplug, and then plug back in the power line). *Check:* In the DOC, the phone was powered up before the Wi-Fi bridge was set up. Ensure the W-Fi Bridge is installed and operational, and cycle power on the phone to reboot the phone (unplug, and then plug back in the power line).

7 PBX Configuration

7.1 Introduction

Ref: http://en.wikipedia.org/wiki/Asterisk_(PBX)

Asterisk is a software implementation of a telephone private branch exchange (PBX). Like any PBX, it allows attached telephones to make calls to one another, and to connect to other telephone services, such as the public switched telephone network (PSTN) and Voice over Internet Protocol (VoIP) services.

More details than you need

The Asterisk software includes many features available in proprietary PBX systems: voice mail, conference calling, interactive voice response (phone menus), and automatic call distribution.

Asterisk supports a wide range of Voice over IP protocols, including the **Session Initiation Protocol (SIP)**, the Media Gateway Control Protocol (MGCP), and H.323. Asterisk can interoperate with most SIP telephones, acting both as registrar and as a gateway between IP phones and the PSTN. The Inter-Asterisk eXchange (IAX2) protocol, RFC 5456, native to Asterisk, provides efficient trunking of calls among Asterisk PBXes, in addition to distributed configuration logic, and call completion to VoIP service providers who support it. Some telephones support the IAX2 protocol directly (see Comparison of VoIP software for examples).

What is SIP

SIP is the **Session Initiation Protocol.** In IP and traditional telephony, network engineers have always made a clear distinction between two different phases of a voice call. The first phase is "call setup," and includes all of the details needed to get two telephones talking. Once the call has been setup, the phones enter a "data transfer" phase of the call using an entirely different family of protocols to actually move the voice packets between the two phones. In the world of VoIP, SIP is a call setup protocol that operates at the application layer.

7.2 Config Files

There are essentially 2 files that control the operation of the phone system.

- Sip.conf – this file is the Configuration file for Asterisk SIP channels, for both inbound and outbound calls.
- Extensions.conf – this file contains the "dial plan" of Asterisk, the master plan of control or execution flow for all of its operations. It controls how incoming and outgoing calls are handled and routed. This is where you configure the behavior of all connections through your PBX.

These files are typically edited directly on the Van server.

File: sip.conf

```
pi@raspberrypi ~ $ cd /etc/asterisk/
pi@raspberrypi /etc/asterisk $ cat sip.conf
[general]
context=default         ; Default context for incoming calls
port=5060               ; UDP port to bnd to (SIP std port=5060)
bindaddr=0.0.0.0        ; IP address to bind to (0.0.0.0 binds to all)
srvlookup=yes           ; Enable DNS SRV lookups on outbound calls

[121]
type=friend             ; both send and receive calls from this peer
host=dynamic            ; this peer will register with us
username=121
secret=z121sip          ; password
canreinvite=no          ; don't send SIP re-invites
nat=yes                 ; always assume peer is behind a NAT
context=phones          ; send calls to 'phones' context
dtmfmode=rfc2833        ; set dtmf relay mode
allow=all               ; allow all codecs

[122]
type=friend
host=dynamic
username=122
secret=z122sip
canreinvite=no
nat=yes
context=phones
dtmfmode=rfc2833
allow=all

[CUPEOC]
type=friend
host=dynamic
secret=eoc
context=phones

[CUPVAN]
type=friend
host=dynamic
secret=van
context=phones

[CUPSITL]
type=friend
host=dynamic
secret=sitl
context=phones
pi@raspberrypi
```

File: extensions.conf

```
pi@raspberrypi /etc/asterisk $ cat extensions.conf
[general]
static=yes              ; default values for changes to this file
writeprotect=no         ; by the Asterisk CLI

[globals]
; variables go here

[default]
; default context

[phones]
; context for our phones
exten => 121,1,Dial(SIP/121)
exten => 122,1,Dial(SIP/122)

exten => CUPVAN,1,Dial(SIP/CUPVAN,20)
exten => 826,1,Dial(SIP/CUPVAN,20)
exten => CUPEOC,1,Dial(SIP/CUPEOC,20)
exten => 362,1,Dial(SIP/CUPEOC,20)
exten => CUPSITL,1,Dial(SIP/CUPSITL,20)
exten => 7485,1,Dial(SIP/CUPSITL,20)

exten => AG6GR,1,Dial(SIP/AG6GR,20)
exten => 647,1,Dial(SIP/AG6GR,20)

; Conference Room
exten => 1111,1,Answer()
exten => 1111,2,ConfBridge(1111)

exten => 500,1,Answer()
exten => 500,2,Playback(demo-echotest)  ; Let them know what is going on
exten => 500,3,Echo
exten => 500,4,Playback(demo-echodone)
exten => 500,5,Hangup

pi@raspberrypi /etc/asterisk $
```

8 Giving the Raspberry Pi a Static IP Address

REF: https://www.modmypi.com/blog/tutorial-how-to-give-your-raspberry-pi-a-static-ip-address
Jul 19, 2013

How to Give your Raspberry Pi a Static IP Address
To log in to your Raspberry Pi remotely, you'll need the IP of the Raspberry Pi – this is basically like your house address and tells the host computer where to look for it on the network. By default, the Raspberry Pi will be given an IP automatically by the router (called Dynamic IP and denoted by DHCP) when you connect to a network. However, this can change whenever you remove the Pi from the network e.g. turn it off.

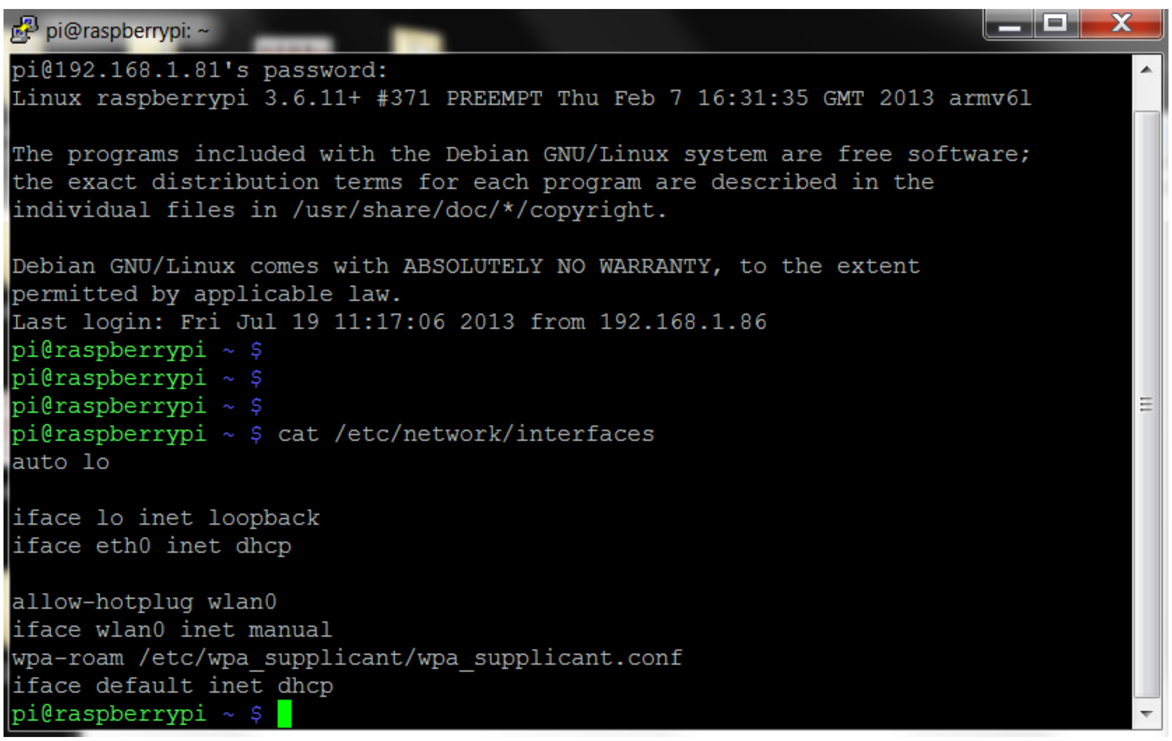
Having a static IP isn't essential, however it will make repeated access to the Raspberry Pi via SSH much simpler, as you'll always know that the Raspberry Pi has the same address. Imagine how much trouble your postman would have if your house constantly changed location :)

8.1 Checking Set Up

Boot into Raspian and log in (Username. pi, Password. raspberry), this will all be command line stuff, so no need to log in to the GUI.

First, we need to list the network interface we currently have available:

cat /etc/network/interfaces



The line . . . .

`iface eth0 inet dhcp`

Implies that we're currently getting out IP address via DHCP, meaning it's being dynamically registered by the router. This is what we want to change!

8.2 Gathering Information

First of all we need to grab some information from our router and Pi. There's a couple of command we need to run to get this info. Have a pen and paper handy! . . .

`ifconfig`

```
pi@raspberrypi: ~
pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:b3:fc:2e
          inet addr:192.168.1.81  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4078 errors:0 dropped:0 overruns:0 frame:0
          TX packets:256 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:264593 (258.3 KiB)  TX bytes:31343 (30.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1104 (1.0 KiB)  TX bytes:1104 (1.0 KiB)

wlan0     Link encap:Ethernet  HWaddr 00:0f:54:12:15:97
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

pi@raspberrypi ~ $
```

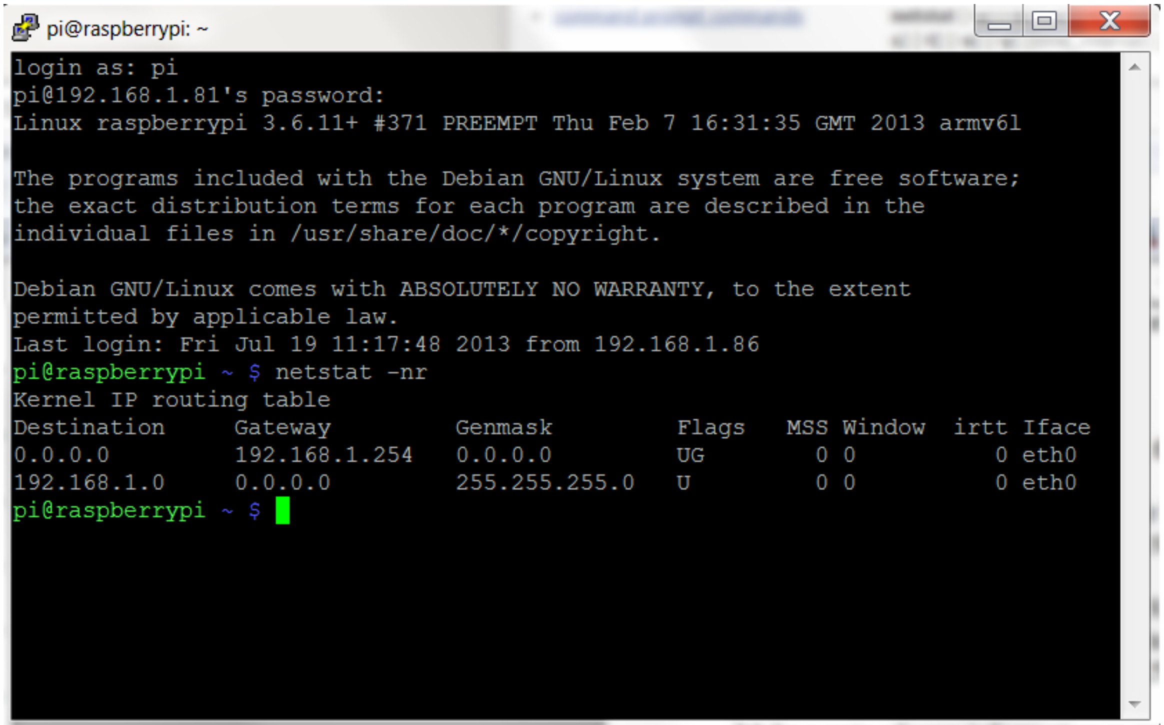
This reveals your router information, the bit you want is after eth0 (the ethernet connection). . . .

```
eth0  Link encap:Ethernet HWaddr b8:27:eb:b3:fc:2c
      inet addr:192.168.1.81 Bcast:192.168.1.255 Mask:255.255.255.0
```

Write down the following information. . .
inet addr – 192.168.1.81 (Pi's Current IP Address)
Bcast – 192.168.1.255 (The Broadcast IP Range)
Mask – 255.255.255.0 (Subnet Mask Address)

We need a little more information before we proceed. Use the command. . .
netstat –nr

(route -n will give you the same info.)



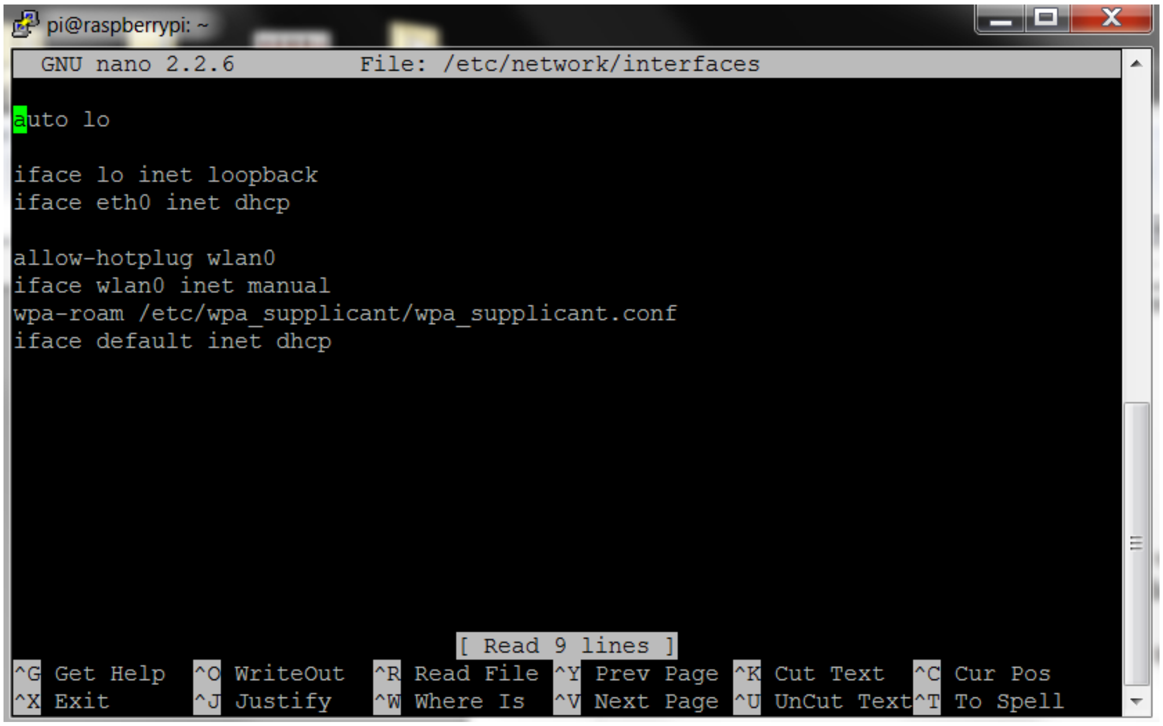
```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.81's password:
Linux raspberrypi 3.6.11+ #371 PREEMPT Thu Feb 7 16:31:35 GMT 2013 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jul 19 11:17:48 2013 from 192.168.1.86
pi@raspberrypi ~ $ netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.1.254   0.0.0.0         UG        0 0          0 eth0
192.168.1.0     0.0.0.0         255.255.255.0   U         0 0          0 eth0
pi@raspberrypi ~ $
```

We need:
'**Gateway**' Address – 192.168.1.254
'**Destination**' Address – 192.168.1.0

8.3 Editing Network Configuration

We now need to plug this information into the Pi's network configuration file using a text editor. I always use nano text editor. . .

```
sudo nano /etc/network/interfaces
```



Simply change the line that reads:

`iface eth0 inet dhcp`

to

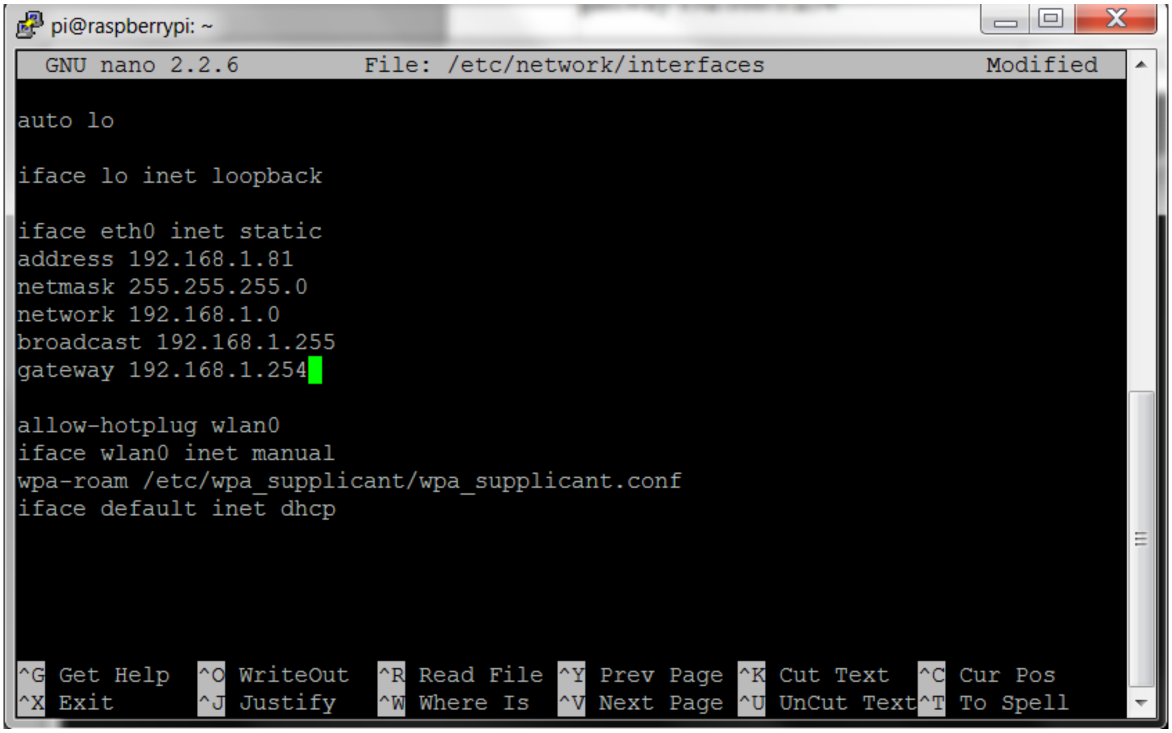
`iface eth0 inet static`

Then directly below this line enter the following (Please Note. **You will need your own addresses we gathered in Part B, more details below**). . . .

```
address 192.168.1.81
netmask 255.255.255.0
network 192.168.1.0
broadcast 192.168.1.255
gateway 192.168.1.254
```

To clarify what each part means. . . .
address – The address you want to give your Pi, this can be any IP in the network range, but it's usually advisable to go higher rather than lower, or you could end up logging different devices to the same IP! I've selected 192.168.1.81, as we're already registered to that address (denoted by '**inet addr**'), but this can be any IP address from the range192.168.1.1 to 192.168.1.253.

netmask – The '**Mask**' address we wrote down earlier.
network – The router IP address, this is the '**Destination**' Address was found earlier. You can also grab this off your router, it will say on the side somewhere.
broadcast – The '**Bcast**' address we wrote down earlier.
gateway – This is the '**Gateway**' address we found earlier.



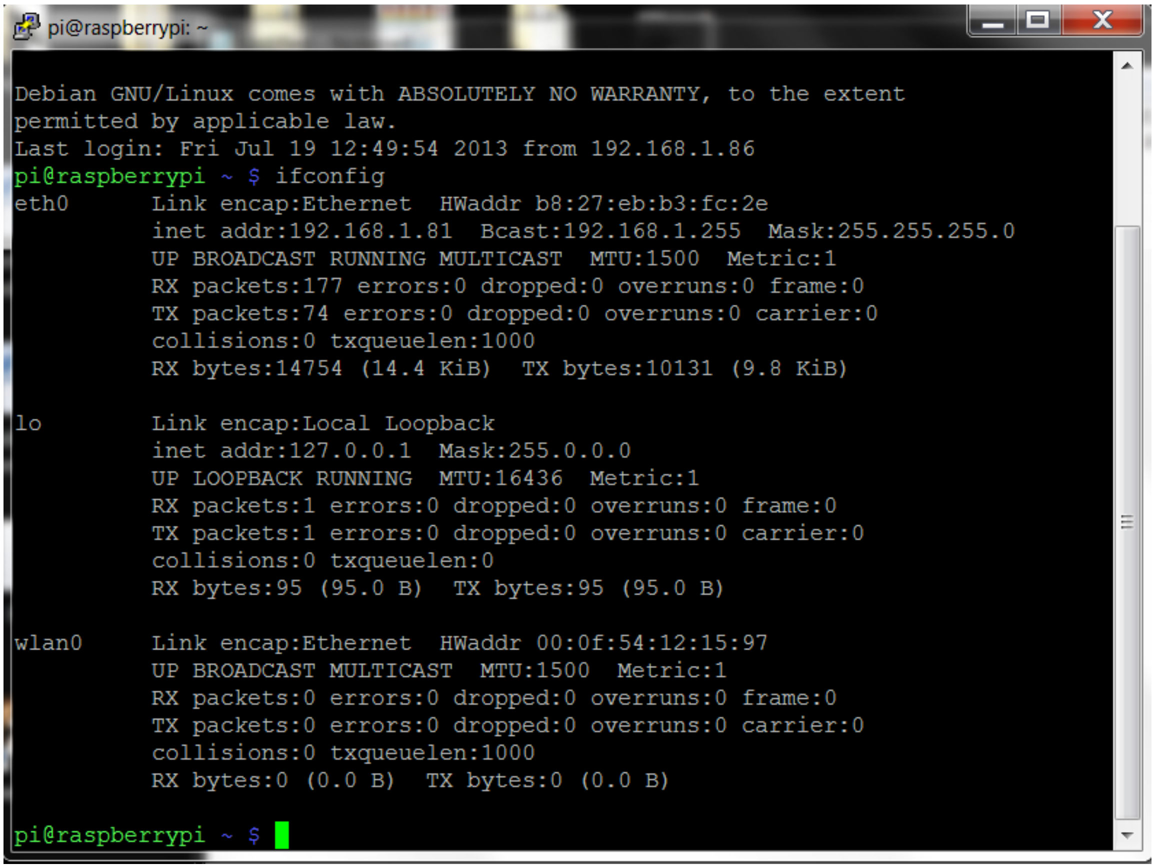
So, it should look something like the above, but with your values! Remember to save before exit, CTRL+X (exit) then yes to save changes!

8.4 Re-check Static IP Configuration

Then we'll need to reboot and check your changes. . .

sudo reboot

Log back in and run
ifconfig
Which should reveal your new settings. .



```
pi@raspberrypi: ~

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Jul 19 12:49:54 2013 from 192.168.1.86
pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:b3:fc:2e
          inet addr:192.168.1.81  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:177 errors:0 dropped:0 overruns:0 frame:0
          TX packets:74 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14754 (14.4 KiB)  TX bytes:10131 (9.8 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:95 (95.0 B)  TX bytes:95 (95.0 B)

wlan0     Link encap:Ethernet  HWaddr 00:0f:54:12:15:97
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

pi@raspberrypi ~ $
```

To double checks all is working as it should, ping your '**Gateway**' Address. . .

ping 192.168.1.254 -c 10

(the -c 10 command simply denotes that you want to ping it 10 times, if you forget to add this, it will ping the address continuously. To stop it press CTRL+C)

```
pi@raspberrypi: ~
pi@raspberrypi ~ $ ping 192.168.1.254 -c 10
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data.
64 bytes from 192.168.1.254: icmp_req=1 ttl=255 time=3.51 ms
64 bytes from 192.168.1.254: icmp_req=2 ttl=255 time=2.49 ms
64 bytes from 192.168.1.254: icmp_req=3 ttl=255 time=3.25 ms
64 bytes from 192.168.1.254: icmp_req=4 ttl=255 time=3.61 ms
64 bytes from 192.168.1.254: icmp_req=5 ttl=255 time=3.43 ms
64 bytes from 192.168.1.254: icmp_req=6 ttl=255 time=4.30 ms
64 bytes from 192.168.1.254: icmp_req=7 ttl=255 time=3.00 ms
64 bytes from 192.168.1.254: icmp_req=8 ttl=255 time=2.75 ms
64 bytes from 192.168.1.254: icmp_req=9 ttl=255 time=2.97 ms
64 bytes from 192.168.1.254: icmp_req=10 ttl=255 time=3.01 ms

--- 192.168.1.254 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 2.499/3.238/4.309/0.488 ms
pi@raspberrypi ~ $
```

This should ping successfully and all packets should be received. If something's not right double check through all your IP addresses, and make sure you're pinging the right address too.

Remember you can always revert back to DHCP by reversing the steps. The 'network' router IP address is sometimes a little fiddly, so check that if you're still having issues!

Hopefully however, your Raspberry Pi is now set up with a static IP address!