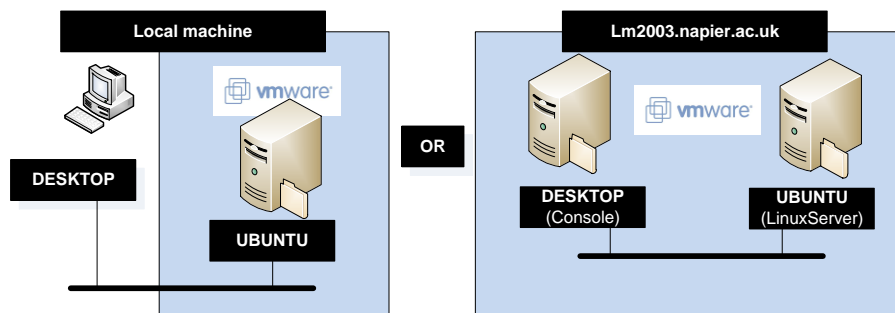


Week	Date	Teaching	Attended
2	Jan 2013	Lab 1: Linux Services/Toolkit Dev't	

Aim: The aim of this lab is to investigate the discovery and configuration of services within Linux. It uses a Linux Ubuntu Virtual Machine (**UBUNTU**). You can either run VMWare Workstation on the local machine in the lab, or connect via a web browser to the VM2003 IaaS virtualisation servers. The Console (**DESKTOP**) on the local machine will be from Windows 7, where in VM2003 it is an Ubuntu console, as outlined below.



Time to spend on practicals:

Up to 4/5 hours (Two supervised hours in the lab, and two/three additional hours, unsupervised).

Activities:

- **Complete Lab 1:** Linux Services/Toolkit Development 1 .pdf from WebCT or <http://www.dcs.napier.ac.uk/~cs342/CSN10102/Lab1.pdf>
- Take **End of Unit Tests** for unit1: <http://asecuritysite.com/security/tests/tests?sortBy=sfc07>

Learning activities:

At the end of these activities, you should understand:

- How to review/define services in Linux.
- How to interface to WinDump from the toolkit development

Reflective statements (end-of-exercise):

- What are the key Linux commands used to discover the services which are being run?
- What is the key folder location for the Web server in Linux?
- Why does Linux need the VNC Client, when Windows uses the Remote Desktop Client?

Source code used:

<http://buchananweb.co.uk/toolkit.zip>

Lab 1: Linux Services

Rich Macfarlane, Bill Buchanan 2013

1.1 Details

Aim: To provide a foundation in setup, discovery, and use of Linux services.

1.2 Activities

An overview of Linux commands, to assist with the lab, can be found at:
<http://www.computerhope.com/unix/overview.htm>

This part of the lab has two elements: the **host** machine (**DESKTOP**) and the Linux virtual server **guest** machine (**UBUNTU**), as shown below. The lab can be completed on either of the 2 architectures shown below. **Either** using VMWare Workstation on the local machine in the lab (shown in Figure 1), or remotely on our VM2003 virtualisation server cluster, via a web browser (shown in Figure 2).

The local lab architecture is shown below. This requires the Linux VM Server to be run using VM Workstation on the local PC.

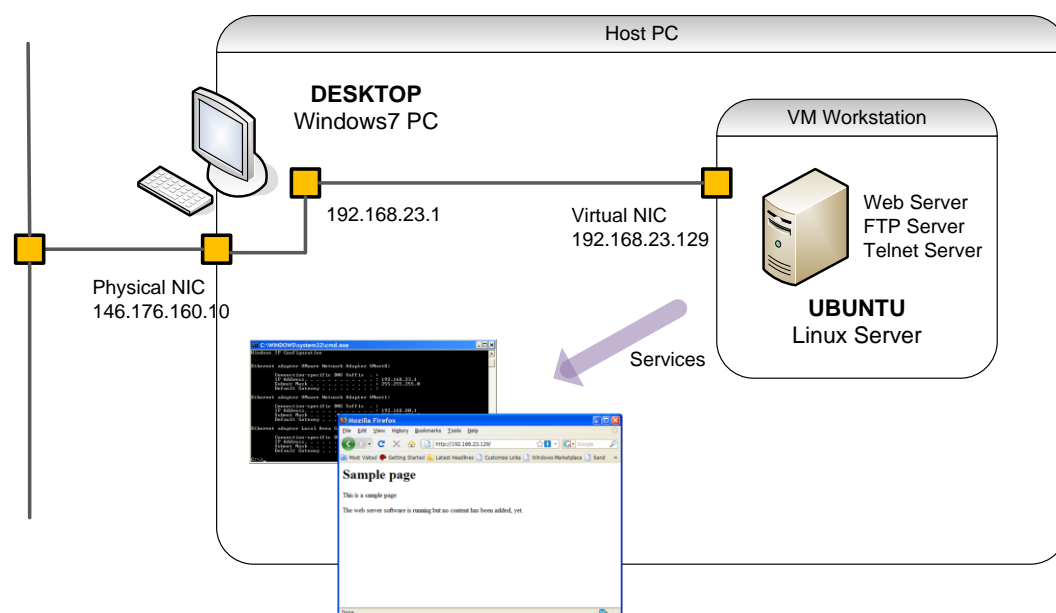


Figure 1 - Lab Architecture

The following video can be used as a guide to complete the local version of the lab:

On-line video demo of the local lab:
http://buchananweb.co.uk/adv_security_and_network_forensics/unix/unix.htm

The virtualisation server cluster lab architecture is shown in Figure 1 - Lab Architecture below. This requires a Linux VM Console and a Linux VM Server to be run in the Virtualisation Cluster (our Private Cloud).

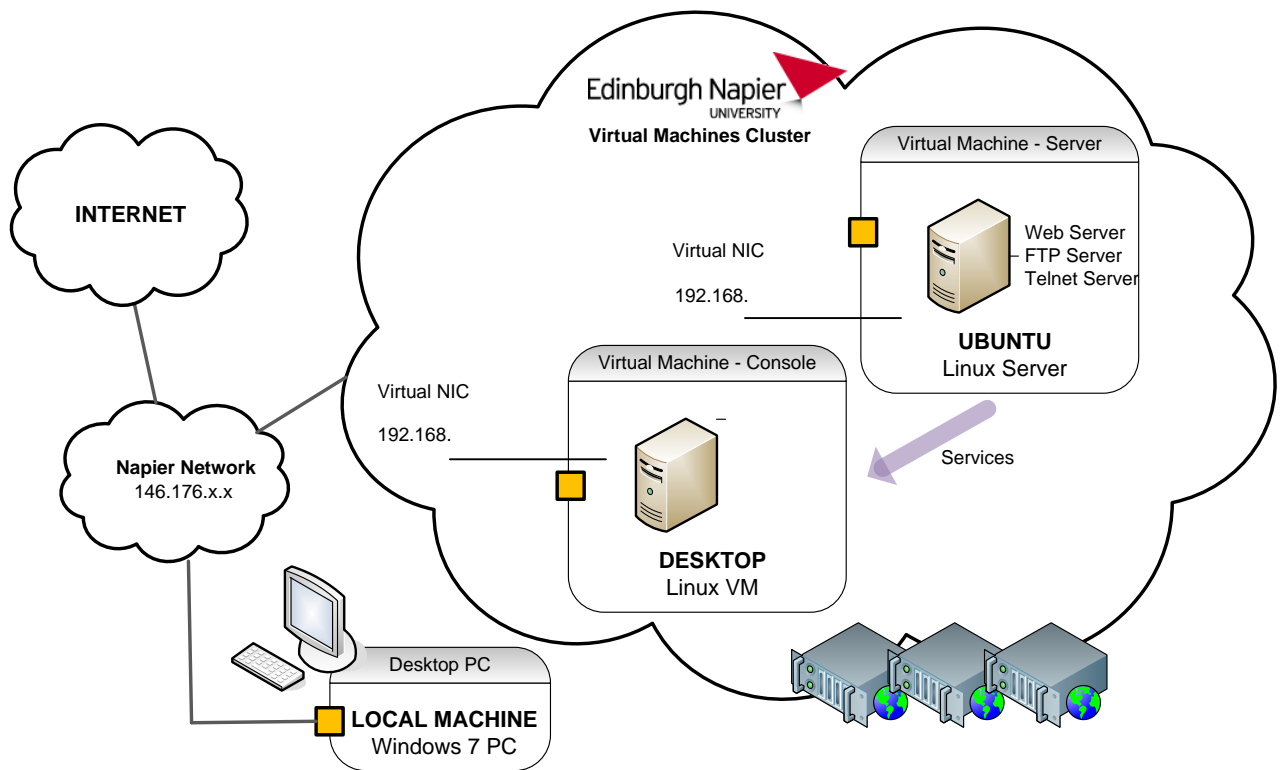


Figure 2 - Cluster Lab Architecture

The following video can be used as a guide to complete the server cluster version of the lab:

On-line video demo of the cluster lab:
http://buchananweb.co.uk/e_presentations/vmware_lab01/vmware_lab01.html

Linux Ubuntu Server

L1.1 Run the virtual Linux server image (locally run the .vmx file, and power on the virtual machine, or deploy your LinuxServer VM in the Adv Security Workspace on the cluster). Log into the server as User name: **Napier**, Password: **napier123**.

When using VM Workstation some shortcuts are worth remembering, such as **CTRL+G** to grab the input focus to the current guest VM, and **CTRL+ALT** to return focus to the host machine.

The following link has a list of shortcuts for VM Workstation:
http://www.vmware.com/support/ws5/doc/ws_learning_keyboard_shortcuts.html

Within the virtual server UBUNTU, open a **Terminal Window** (command line window), from **Applications->Accessories->Terminal** such as shown in Figure 3.

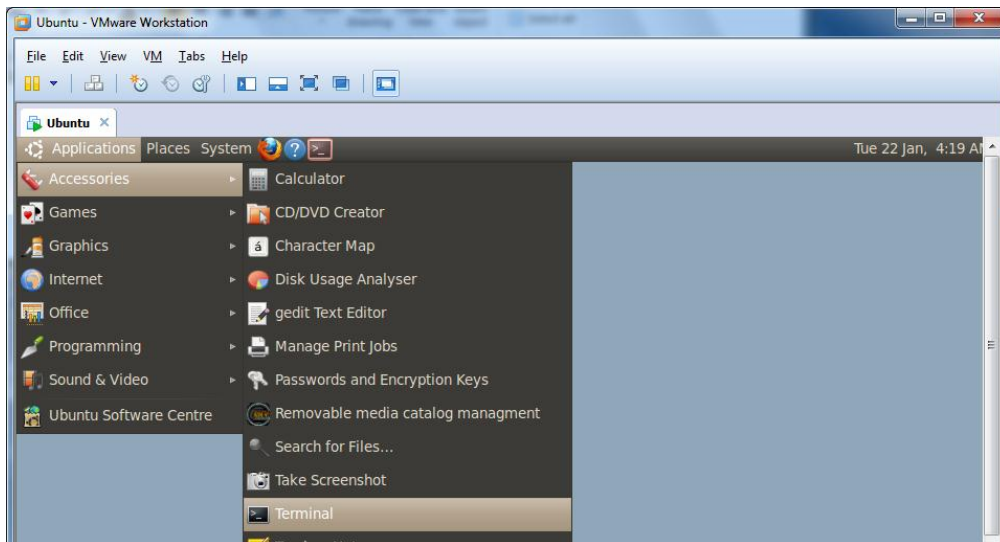
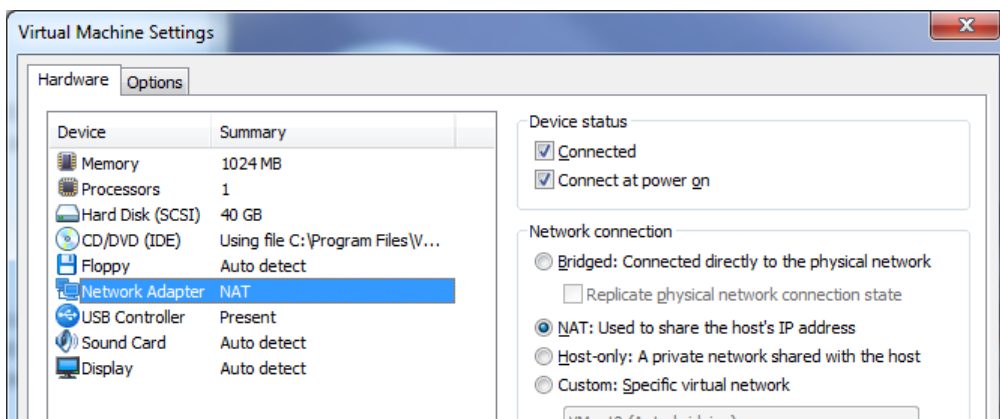


Figure 3 - Run Linux Terminal

From **UBUNTU** determine the servers IP address using the Linux **ifconfig** command.

If your **UBUNTU** VM does not have an IP Address, check that it is connected to the host via NAT, using the menu **VM>Settings**, as shown below, and issue the **dhclient** command to get an IP Address from the VMWare DHCP server.



From **DESKTOP** open a command line window and determine the IP Address of the host PC using the Windows **ipconfig|more** command (or **ifconfig|more** if using the cluster). Look for the interface which is on the same subnet as UBUNTU.

Complete the IP Addressing diagrams in Figure 4 **OR** 5, **depending on which architecture you are using**. Fill in the IP addresses(s) of the DESKTOP machine, and the UBUNTU virtual server.

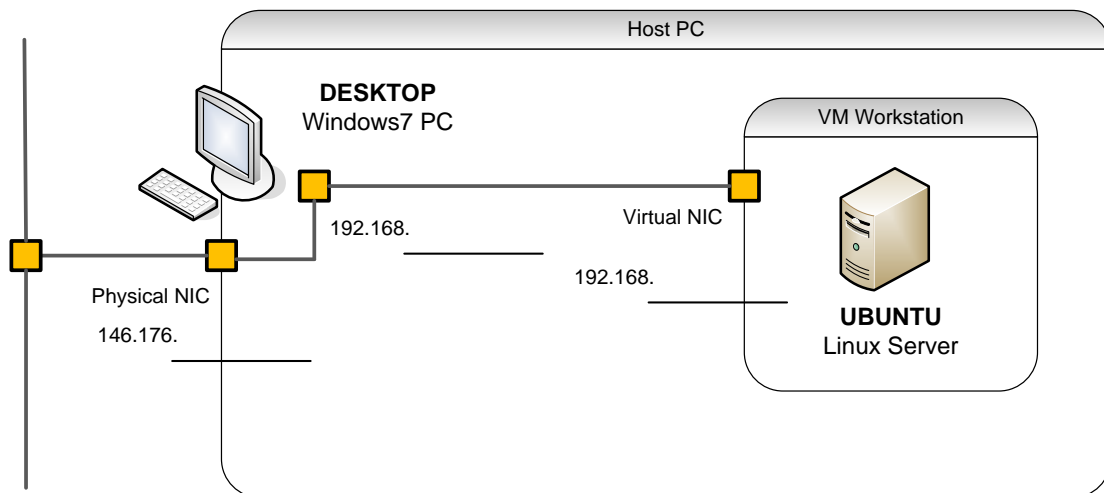


Figure 4 - Local lab IP Addressing

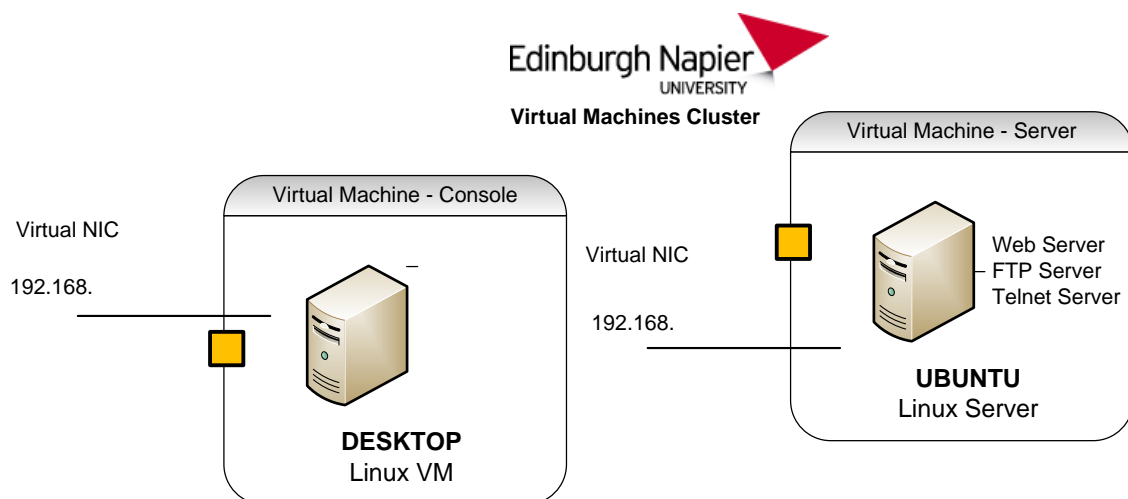


Figure 5 - Cluster IP Addressing

L1.2 From DESKTOP, **ping** UBUNTU, and vice-versa, to check the connectivity.

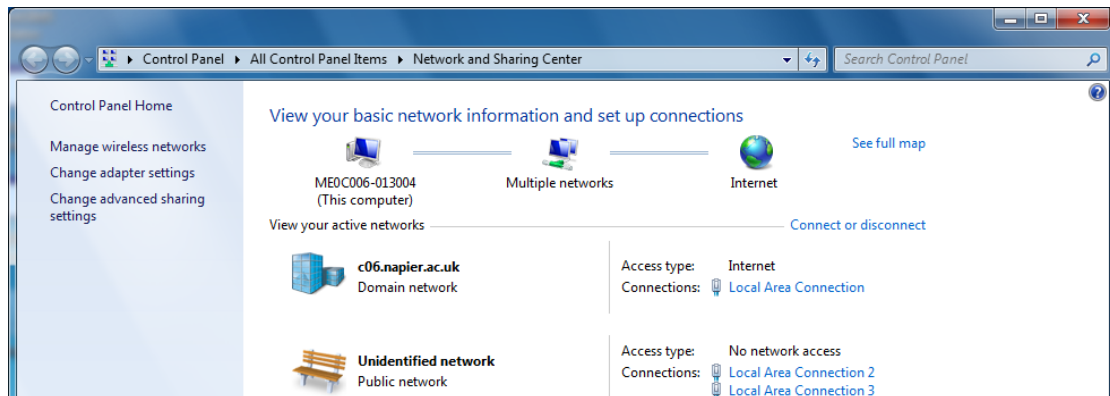
☞ Was the ping from DESKTOP to UBUNTU successful?	YES/NO
☞ Was the ping from UBUNTU to DESKTOP successful?	YES/NO
☞ Why might this be?	

Windows 7 Firewall

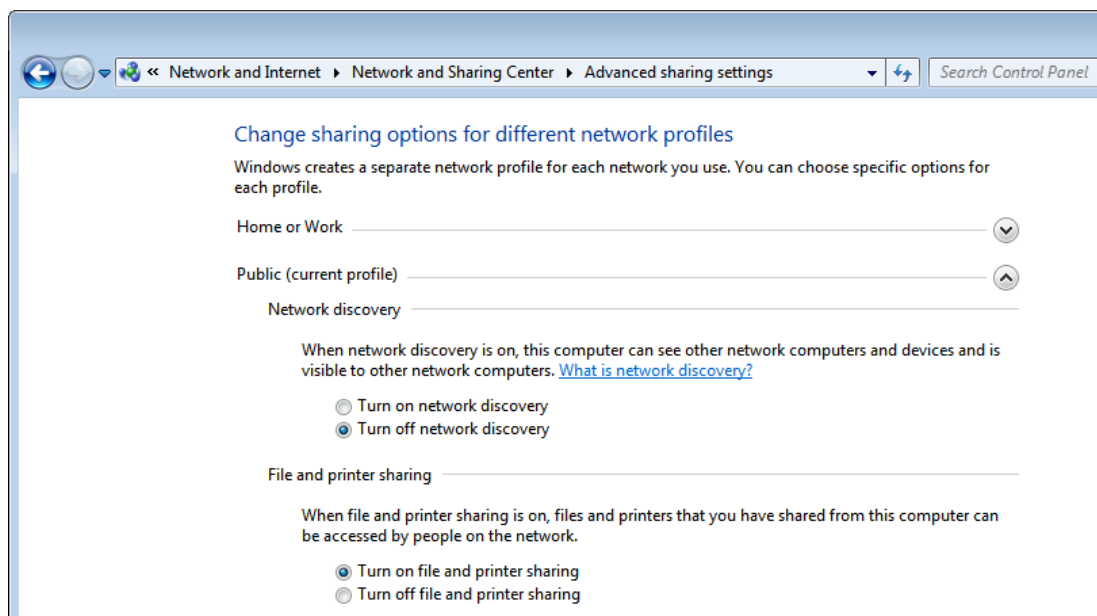
If DESKTOP is a Windows7 machine it may be that the firewall is blocking virtual networks (typically connected as network type **Public**), and so we need to either allow ICMP traffic through, or turn the firewall off on the VMWare network. (192.168.x.x)

On DESKTOP, open the Windows Network and Sharing Center window. This can be accessed by left clicking the **Windows Network Connections Icon**, (📶 or 🖨️) in the

notification area, and selecting **Open Network and Sharing Center**. (It can also be accessed via the Windows Control Panel).



Check which type of network UBUNTU is connected to (typically **Public**). Select **Change advanced sharing settings**, and **Turn on file and printer sharing** for this network type, as shown below.



This should allow ICMP packets used by the ping tool through the firewall for this network type.

Retest the connectivity test from WINDOWS2003 to DESKTOP.

 Was the ping from UBUNTU to DESKTOP successful?	YES/NO
---	--------

Linux Services

L1.3 To list the running network services in the UBUNTU server, the **netstat** command can be used. From UBUNTU use **netstat --help** to check the arguments and options.

Use the **netstat -ltu** command to determine the services that are running. (-l listening, -t TCP, -u UDP)

☞ List some of the services (and their port numbers) which are running on the server:

Note: The **-n** flag can be used to find the numeric port numbers of the listening servers. The IANA Port Numbers web page lists the official services and their protocol/portnumbers. Try googling for it.

To view the associated processes for the services, on **UBUNTU** use the **-p** flag such as:

```
sudo netstat -antp
```

☞ List some of the services and their processes?

Services: Web

L1.4 In the virtual Linux server **UBUNTU**, open a Terminal Window and navigate to the folder **/var/www**. Use the **ls** command to list the contents of the directory.

☞ What are the names of the files in this directory, and what type of files are they:

L1.5 From the host system **DESKTOP**, using a browser, connect to the Web Server running on **UBUNTU** using **http://w.x.y.z**, where **w.x.y.z** is the IP address of your **UBUNTU** Linux server, as shown below.

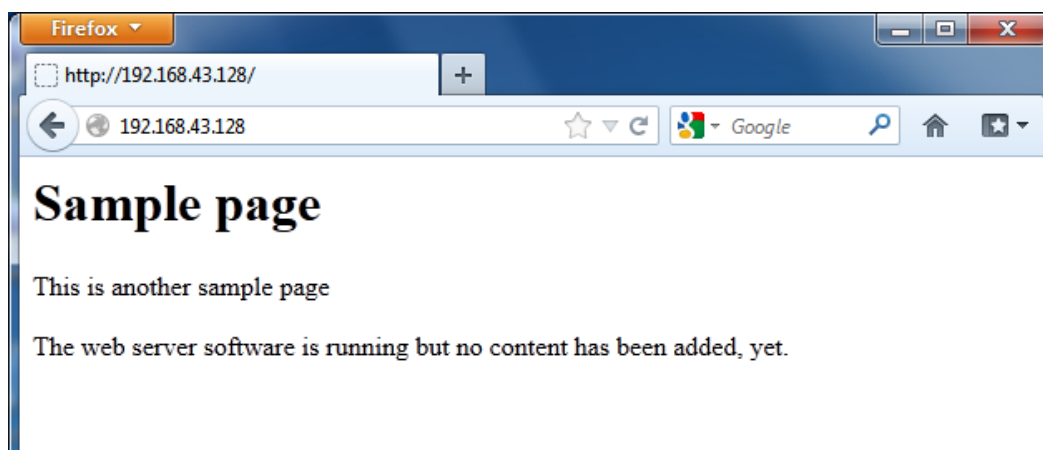


Figure 6 - HTTP connection to web server on **UBUNTU**

On UBUNTU use the **netstat -atu** command to determine the services that are running. (-a all connections, listening and established, -t TCP, -u UDP)

- ☞ Can you see the established client/server connection between DESKTOP and UBUNTU?
- ☞ What is the client port?

From DESKTOP, connect to the Web Server, but this time using telnet from a windows command shell or Putty (you may need to activate the Windows telnet client via **Control Panel>Programs & Features>Turn Windows Features On & Off**):

```
telnet w.x.y.z 80
```

and then send the HTTP GET command to the server:

```
GET /index.html
```

- ☞ What is the response from the web server?
- ☞ How does this relate to accessing the home page, using the web browser?

L1.6 On the UBUNTU Linux Server create your own home web page.

Open an editor to create a new html web page, using a command such as **sudo vi** or **sudo gedit**.

Save the page with a name such as **my_home_page.html**, and in the **/var/www** folder . It should contain a link to the default page (index.html), such as:

My Home Page

This is a sample HTML page. Click [here](#) to return to the default home file.

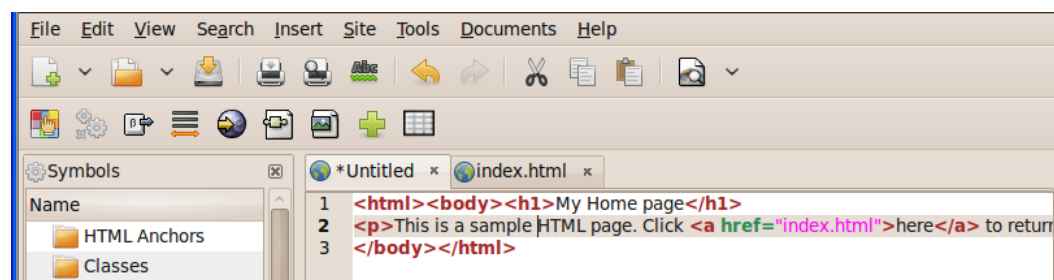
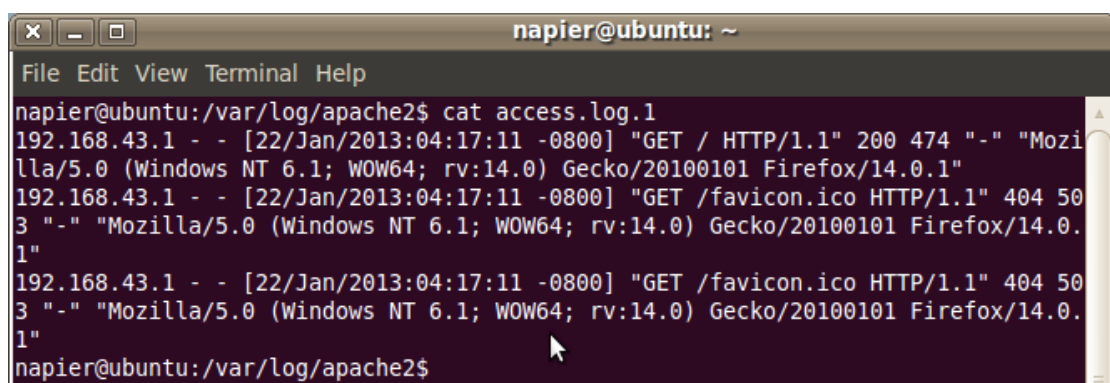


Figure 7 - Creating 'My Home Page' web page

- ☞ Can you access this new web page, via a browser, from the host (DESKTOP)?
YES/NO
- ☞ On UBUNTU, go to `/var/log/apache2`. What are the contents of the folder?
- ☞ Using the `cat filename` command or an editor such as `vi` find out, and explain, what the different files contain:
- ☞ How might these log files be used to trace malicious activity?



```
napier@ubuntu: ~  
File Edit View Terminal Help  
napier@ubuntu:/var/log/apache2$ cat access.log.1  
192.168.43.1 - - [22/Jan/2013:04:17:11 -0800] "GET / HTTP/1.1" 200 474 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0) Gecko/20100101 Firefox/14.0.1"  
192.168.43.1 - - [22/Jan/2013:04:17:11 -0800] "GET /favicon.ico HTTP/1.1" 404 503 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0) Gecko/20100101 Firefox/14.0.1"  
192.168.43.1 - - [22/Jan/2013:04:17:11 -0800] "GET /favicon.ico HTTP/1.1" 404 503 "-" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0) Gecko/20100101 Firefox/14.0.1"  
napier@ubuntu:/var/log/apache2$
```

Figure 8 - Apache Web Server Access Log

Services: Telnet

L1.7 Connect to the Telnet Server on UBUNTU from DESKTOP using `telnet w.x.y.z`, where w.x.y.z is the IP address of UBUNTU. Login in with the user: **napier** (password: **napier123**).

On UBUNTU, use the `netstat -atu` command to determine the services that are running. (-a all connections, listening and established, -t TCP, -u UDP)

- ☞ Can you see the established telnet client/server connection between DESKTOP and UBUNTU?

☞ What is the default home folder for Telnet on UBUNTU? (Look up/google the Linux command to determine the current [working] directory)

Quit from Telnet, using the **exit** command.

Services: FTP

L1.8 From your host, connect to the FTP Server from DESKTOP, via a web browser using **ftp://w.x.y.z**, where w.x.y.z is the IP address of UBUNTU, as shown in Figure 9.

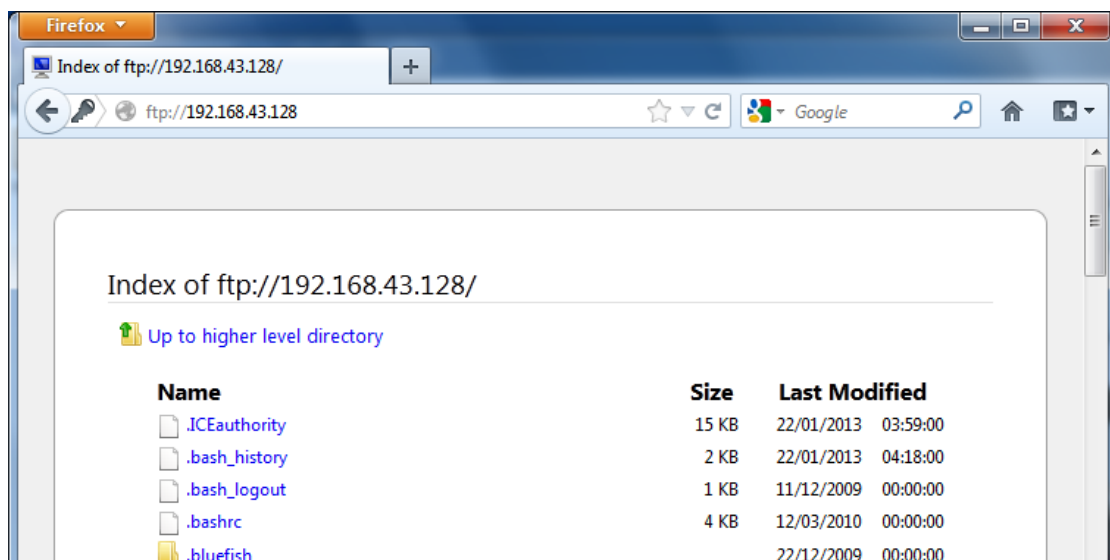


Figure 9 - FTP connection

Open a command line window from DESKTOP, and connect to the FTP Server using the command:

```
telnet w.x.y.z 21
```

Then enter the commands in **bold** (and note the some of the commands that you get beside the sample return ones):

```
USER napier
331 Password required for napier.
PASS napier123
230- Linux ubuntu 2.6.31-14-generic #48-Ubuntu SMP Fri Oct 16 14:04:26 UTC
      2009 i686
230-
230- To access official Ubuntu documentation, please visit:
230- http://help.ubuntu.com/
230-
230 User napier logged in.
HELP
```

214- The following commands are recognized (* =>'s unimplemented).

USER	PORT	STOR	MSAM*	RNTO	NLST	MKD	CDUP
PASS	PASV	APPE	MRSQ*	ABOR	SITE	XMKD	XCUP
ACCT*	TYPE	MLFL*	MRCP*	DELE	SYST	RMD	STOU
SMNT*	STRU	MAIL*	ALLO	CWD	STAT	XRMD	SIZE
QUIT	RETR	MSOM*	RNFR	LIST	NOOP	XPWD	

PWD

257 "/home/napier" is current directory.

TYPE I

200 Type set to I.

PASV

227 Entering Passive Mode (192,168,75,136,146,31)

LIST

🔗 Did the **LIST** command succeed?

YES/NO

The **PASV** FTP command opens up a different port for the data transfer. This is calculated from the last two digits of the Passive Mode response (227 response). It is calculated as, the second last (146) digital multiplied by 256, plus the last digital (31).

So, in this case, it would be:

$$\text{Port} = 146 * 256 + 31 = 37397$$

Next, open up the data transfer channel by creating a new Telnet connection, in a second command line window, such as with the command:

telnet w.x.y.z 37397

Now try the **LIST** command again, in the 1st command window.

🔗 Did the LIST command succeed?

YES/NO

🔗 On UBUNTU, go to **/var/log** dir. View the syslog file (using the **cat syslog** command or the **vi** editor). What is its contents?

🔗 How might these log files be used to trace malicious activity?

🔗 View the contents of **/etc/inetd.conf** file. How is this used to enable services?

Services: Remote Desktop

L1.9 On the host PC, download the **VNC Client** application from:

<http://www.realvnc.com/cgi-bin/download.cgi>

Downloads Tab, and choose Free Edition once you have installed.

Then from the host (DESKTOP), connect to UBUNTU using the VNC Client, as shown in Figure 10.

Note: A dialog may appear on the server, which requires you to allow access to the VNC Client program.

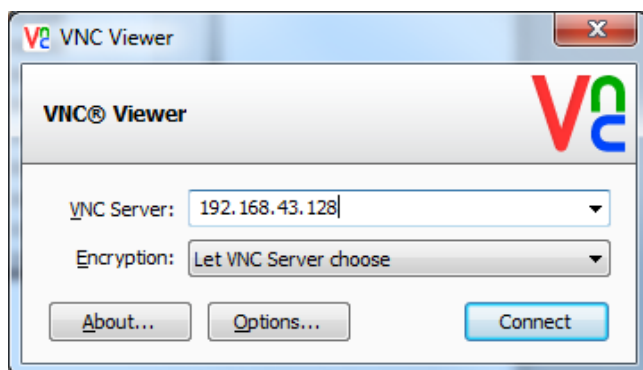


Figure 10 - VNC Viewer Client Application

☞ Which is the service which is running on UBUNTU that allows the remote connection to happen?

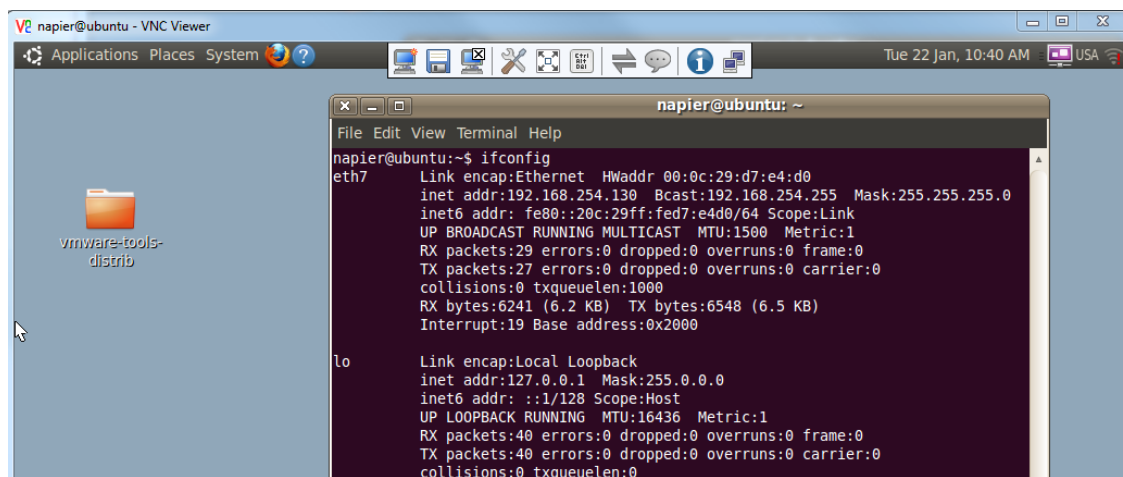
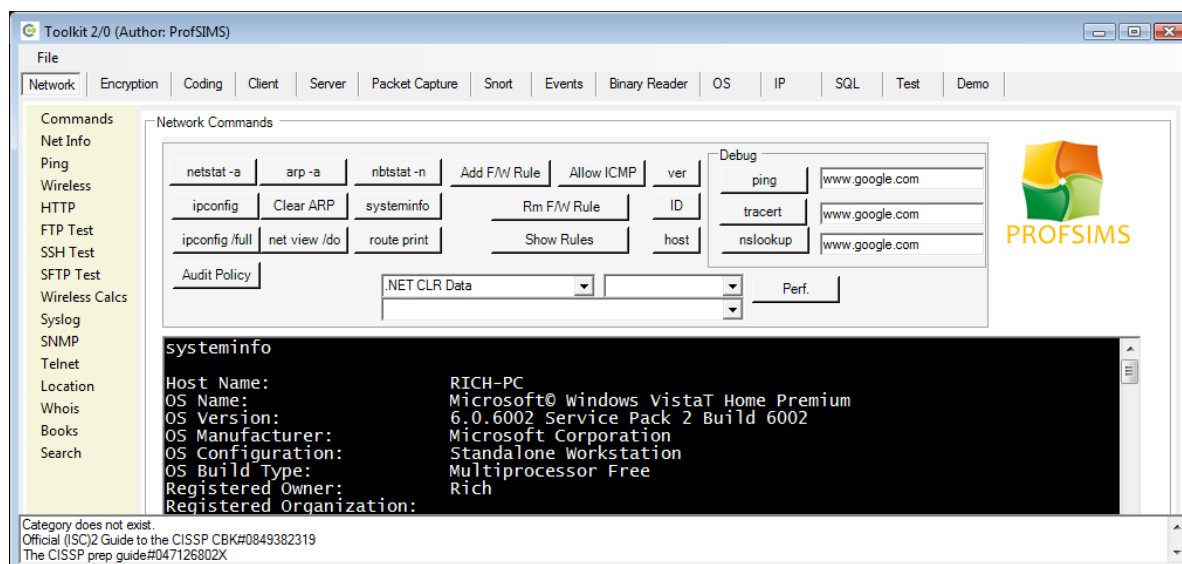


Figure 11 - VNC Viewer Application with Remote Desktop of UBUNTU

1.3 Security & Forensics Toolkit Development

The objective of this series of labs is to build an integrated **Security & Forensic Toolkit** in C#. Partially complete versions of the toolkit, written in C#, will be used as a starting point in the labs. A complete, fully functional toolkit can also be downloaded from the link below.

The finished toolkit application can be downloaded from:
<http://buchananweb.co.uk/dotnetclientserver.zip>



1.4 Toolkit Development 1 - Windows Utilities

This toolkit lab shows how to integrate some Windows command line security utilities in to the toolkit software. This toolkit software development lab has an associated video demo.

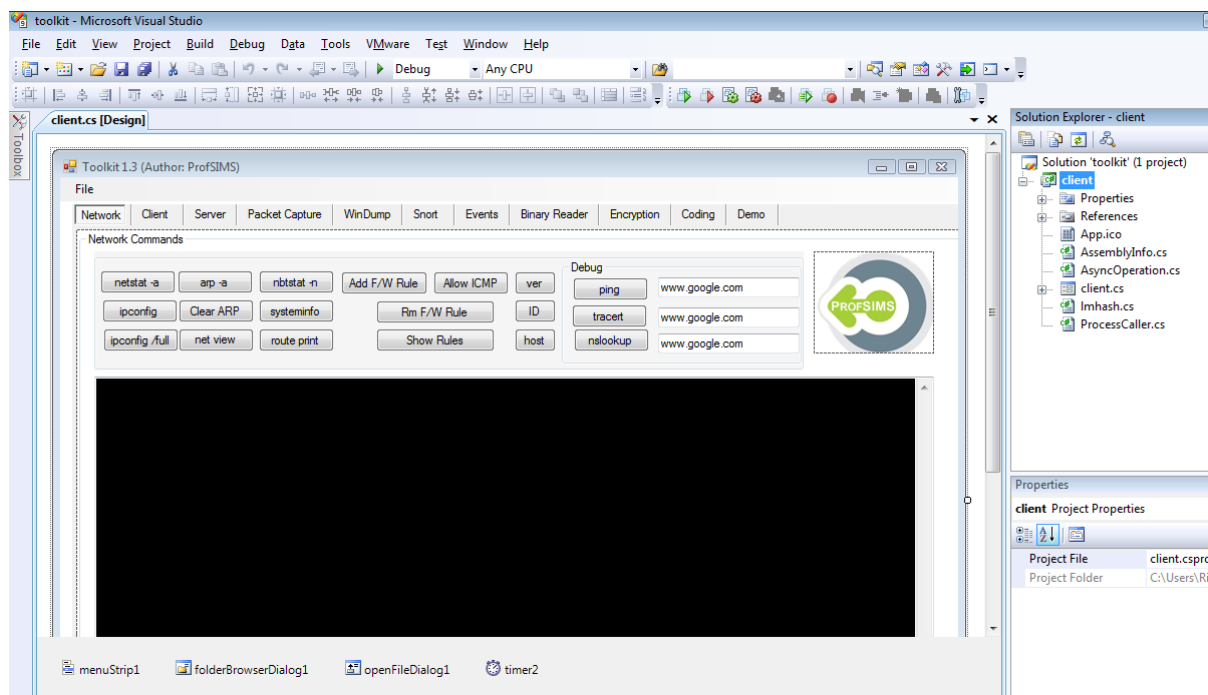
Video demo of toolkit software development part 1:
http://buchananweb.co.uk/adv_security_and_network_forensics/toolkit01/toolkit01.htm

For this lab, download the partially finished toolkit application source code (a Visual Studio C# Solution) from the link below:

Toolkit source code:
<http://buchananweb.co.uk/toolkit.zip>

Extract the source code for the C# Windows Application to a local folder. Next open the toolkit application with Visual Studio (VS) (double click the VS solution file **toolkit.sln**). You should see the **Solution Explorer** panel on the right of the VS Window.

Open the Toolkit Windows Form by double clicking the **client.cs** module, from the Solution Explorer panel. The Toolkit form should now be shown in the panel on the left. The Network tab should be displayed, as shown below.



L1.10 Select the [Network] tab, and to edit the code for the **netstat -a** button, double click it. The event handler code associated with the button should be displayed. Add the following code to the event handler:

```
runProgram("netstat", "-a");
```

Run the program, and test.

L1.11 Select the [Network] tab, and complete the rest of the buttons (see <http://buchananweb.co.uk/dotnetclientserver.zip>, or the figure below for the functions required).

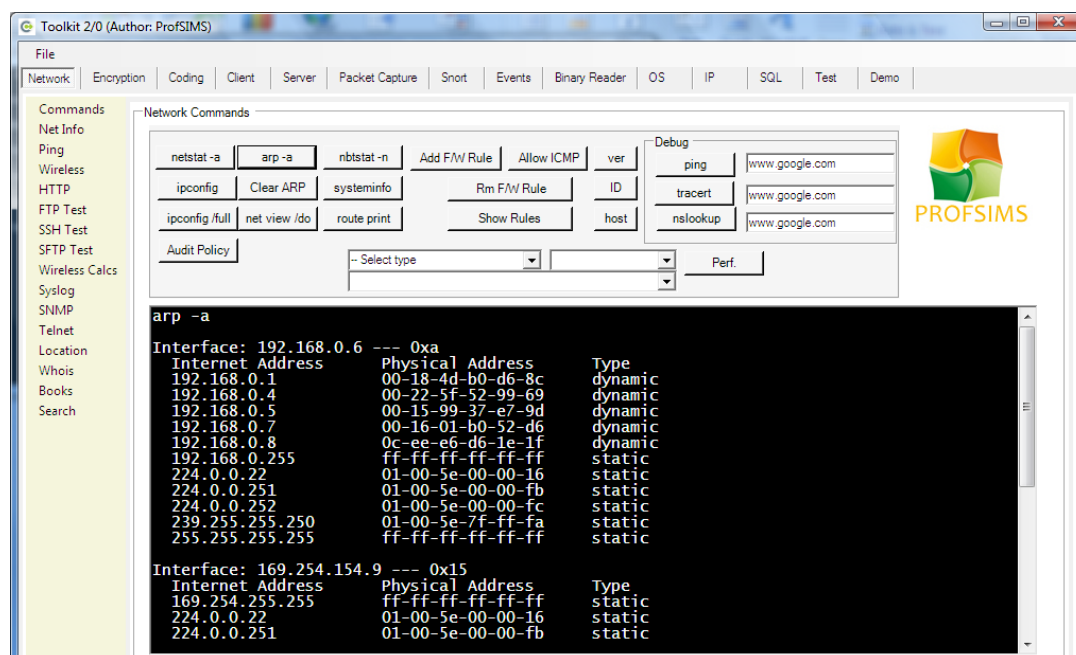


Figure 12 - Buttons to add

L1.12 Select the [Network] table, and complete the rest of the buttons (netstat -a, “arp -a”, “nbstat -n”, “systeminfo”, “ipconfig”, “ipconfig /all”, “route print” and “net view|”). See Figure L1.4.

For Audit Policy add:

```
runProgram("Auditpol", "/get /category:*");
```

For Clear ARP add:

```
runProgram("netsh", "interface ip delete arpcache");
```

For Add Firewall Rule:

```
runProgram("netsh", "advfirewall firewall add rule name=\"NetworkSims Rule\"  
dir=in action=allow protocol=TCP localport=65000");
```

For Add ICMP Rule:

```
runProgram("netsh", "advfirewall firewall add rule name=\"NetworkSims Rule\"  
protocol=icmpv4:8,any dir=in action=allow");
```

For Delete Rule:

```
runProgram("netsh", "advfirewall firewall delete rule name=\"NetworkSims  
Rule\" dir=in");
```

For Show Rules:

```
runProgram("netsh", "advfirewall firewall show rule name=\"NetworkSims  
Rule\"");
```

L1.10 Now add three buttons, and three text boxes (tbPing, tbTracert and tbTracert) and add a ping, tracert and nslookup button. Next add the code to each of the buttons:

```
runProgram("ping", tbPing.Text);  
runProgram("tracert", tbTracert.Text);  
runProgram("nslookup", tbTracert.Text);
```