

Week	Date	Teaching	Attended
4	Feb 2013	Lab 3: Vulnerability Analysis/ Toolkit 3	
<p>Aim: The aim of this lab is to investigate possible vulnerabilities for Windows and Ubuntu, and how to detect signs of intrusions. The lab has been built around using two virtual machines. UBUNTU and a WINDOWS2003 VMs. These can be either hosted on a local PC or run on the Napier virtual environment at vc2003.napier.ac.uk.</p> <p>Time to complete: 4 hours (Two supervised hours in the lab, and two additional hours, unsupervised).</p> <p>Activities:</p> <ul style="list-style-type: none"> • Complete Lab 3: Vulnerability Analysis /Toolkit Development 3 .pdf from WebCT or http://www.dcs.napier.ac.uk/~cs342/CSN10102/Lab3.pdf • Complete the End Of Unit Tutorial Questions for this unit, with the NetworkSims Test Engine, and online questions at: http://asecuritysite.com/security/tests/tests?sortBy=sfc09 <p>Learning activities: At the end of these activities, you should understand:</p> <ul style="list-style-type: none"> • How to detect network scanning for vulnerabilities. • How SQL can be passed through a Web page to the database. • How to use vulnerability tools to perform basic vulnerability testing. • How to integrate Snort/Nmap within the toolkit. <p>Reflective statements (end-of-exercise):</p> <ul style="list-style-type: none"> • Why are tools such as nmap, hping, hydra and so on, seen a malicious, while also being useful to security professionals? • What methods would be applied to defend against SQL Injection attacks? <p>Source code used: http://buchananweb.co.uk/toolkit.zip</p>			

Lab 3: Vulnerability Analysis

3.1 Details

Aim: The aim of this lab is to investigate possible vulnerabilities for Windows and Ubuntu Linux, and how to detect basic signs of intrusions.

3.2 Activities

To assist with this part of the lab and the following demo can be used:
http://buchananweb.co.uk/e_presentations/asfn_lab03_part01/asfn_lab03_part01.html

L3.1 Run the Windows Server 2003 virtual image **WINDOWS2003**. Log into the Windows server as User name: **Administrator**, Password: **napier**).

Within the virtual image WINDOWS2003, run a **Command Prompt Window** and determine the servers IP Address using **ipconfig**.

L3.2 Run the Linux virtual image **UBUNTU**. Log into the Linux server as User name: **napier**, Password: **napier123**).

Within the virtual image UBUNTU, open a **Terminal Window** (Applications->Accessories->Terminal) and determine the servers IP Address using **ifconfig**.

Complete the IP Addressing diagram in Figure , or Figure , depending on which architecture you choose to use.

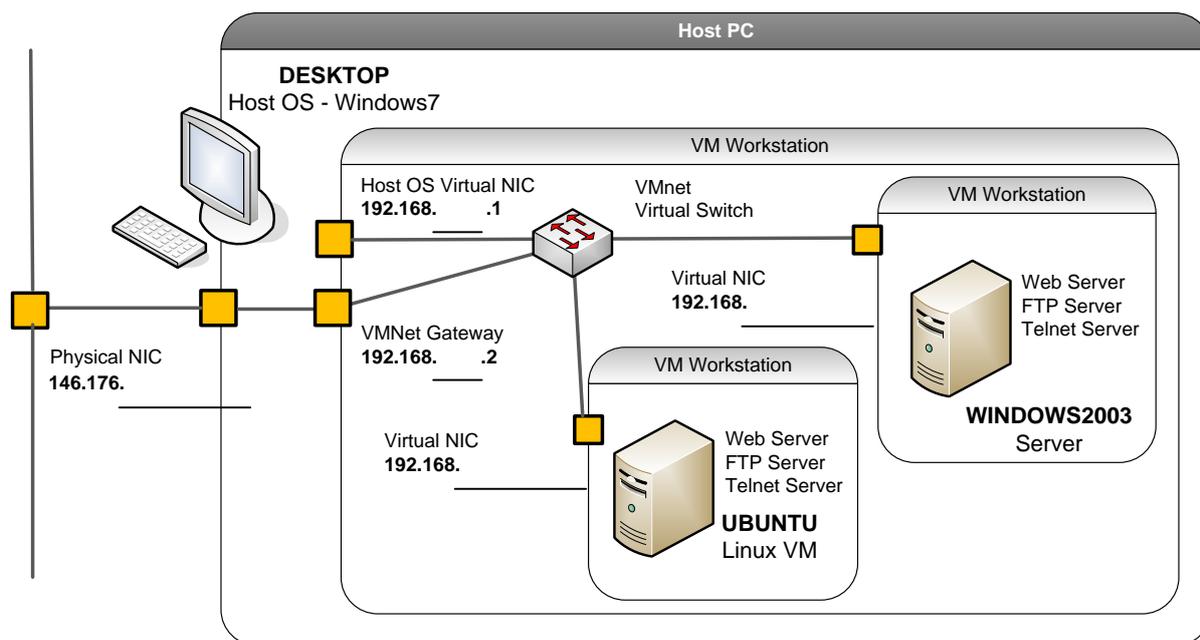


Figure 1 – Local Lab Architecture and IP Addressing

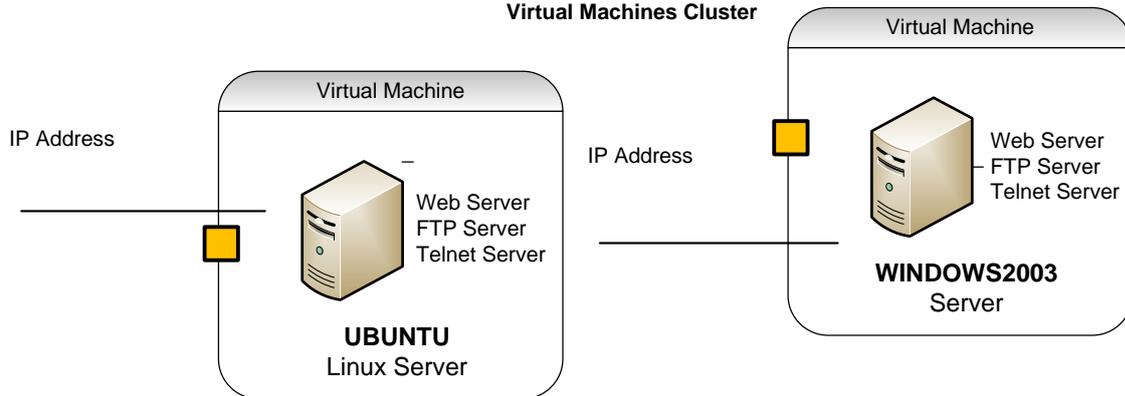
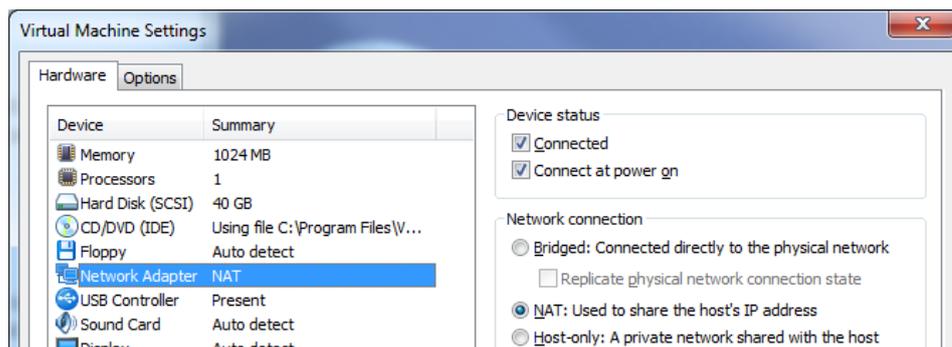


Figure 2 – Cluster-based Lab IP Addressing

If your WINDOWS2003 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 **ipconfig /renew** command (or disable/enable your adapter) to get a new IP Address from the VMWare DHCP server.



Check connectivity between WINDOWS2003 and UBUNTU using **ping**.

☞	Are connectivity tests successful?	YES/NO
---	------------------------------------	--------

3.2.1 Host Network Scanning

L3.3 Using the **nmap** network scanner, from WINDOWS2003 run a **Port Scan** on the UBUNTU machine, and vice-versa with **nmap IPAddress**. (do not scan any other systems)

☞	To assist with the lab and more information on nmap , the manual can be found at: http://www.nmap.org/book/man.html
---	---

Note some of the well known services discovered:

☞	List some of the well known Services running on the WIN2003 Windows Server VM :
---	---

☞ List some of the **Services** running on the **UBUNTU Linux VM**:

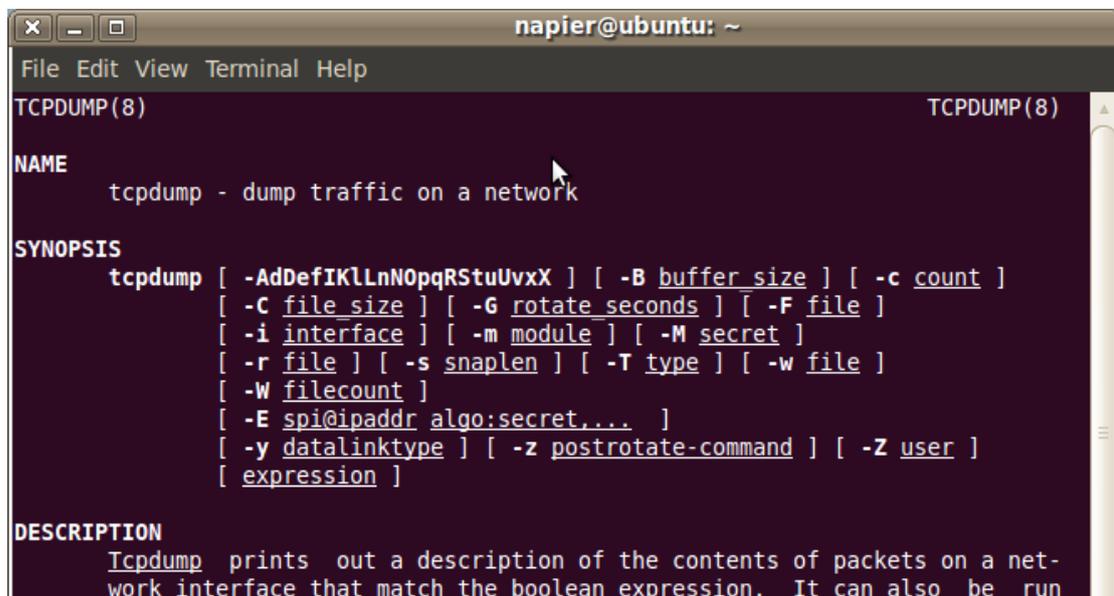
Verify the results using the **netstat** command on both VMs.

3.2.2 Network Scanning Traffic Capture, using Tcpcdump

L3.4 Tcpcdump is a common command line packet capture tool. It uses the libpcap packet capture library on Linux/Unix systems.

☞ More information on **tcpcdump**, can be found at:
<http://www.tcpcdump.org>

From UBUNTU, the **man tcpcdump** command can be used to check the arguments/flags we can use with tcpc dump.



```
napier@ubuntu: ~
File Edit View Terminal Help
TCPDUMP(8) TCPDUMP(8)
NAME
    tcpcdump - dump traffic on a network
SYNOPSIS
    tcpcdump [ -AdDefIKLLnNOpqRStuUvxx ] [ -B buffer_size ] [ -c count ]
             [ -C file_size ] [ -G rotate_seconds ] [ -F file ]
             [ -i interface ] [ -m module ] [ -M secret ]
             [ -r file ] [ -s snaplen ] [ -T type ] [ -w file ]
             [ -W filecount ]
             [ -E spi@ipaddr algo:secret,... ]
             [ -y datalinktype ] [ -z postrotate-command ] [ -Z user ]
             [ expression ]
DESCRIPTION
    Tcpcdump prints out a description of the contents of packets on a network interface that match the boolean expression. It can also be run
```

From UBUNTU, run **sudo /usr/sbin/tcpcdump -i eth1**, and run **nmap** on WINDOWS2003.

☞ What can be observed from **UBUNTU**?

☞ Can the machine performing the scanning be identified by analysing the tcp output?

L3.5 Windump is a windows port of tcpcdump and uses the similar winpcap packet capture library on Windows systems.

From WINDOWS2003, run **windump -i 2**, and run a **Port Scan** using **nmap** on UBUNTU.

If **windump** is not found, one way to find a file from the Windows command line is using the following command:

```
dir c:\ /s /b | find "WinDump"
```

(Change to the dir where it is found, and run it from there)

☞ What can be observed from **WINDOWS2003**:

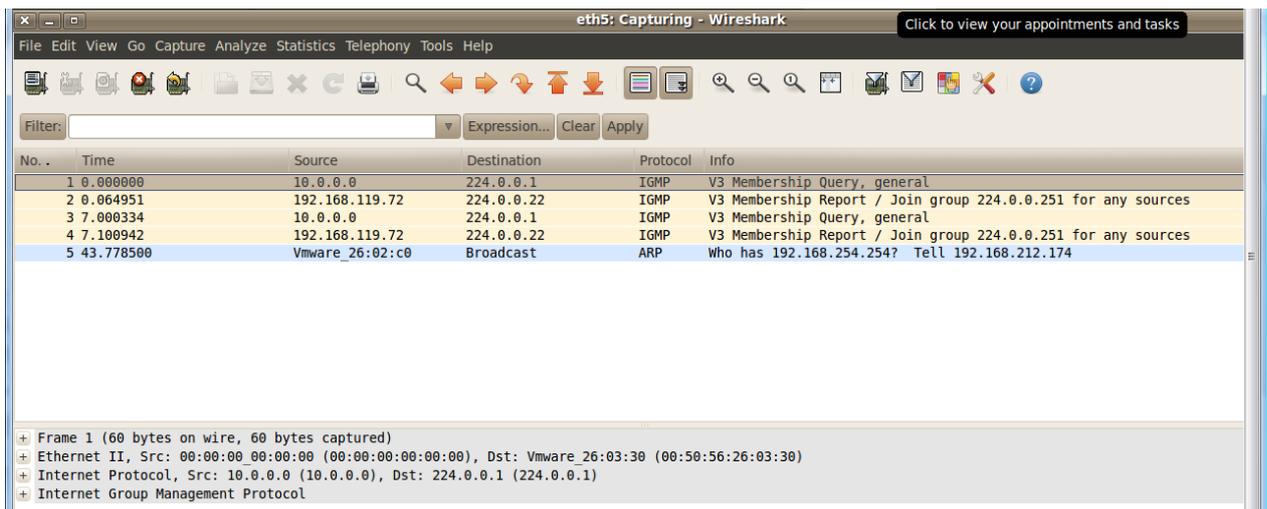
3.2.3 Network Scanning Detection, using an IDS

☞ To assist with this part of the lab and the following demo can be used:
http://buchananweb.co.uk/e_presentations/asfn_lab03_part01/asfn_lab03_part02.html

L3.6 From **UBUNTU**, run the **Wireshark** packet sniffer with the command:

```
sudo wireshark &
```

Start to capture traffic using the **Capture>Interfaces** dialog to check which interfaces have traffic, as shown below.



(you may want to change your screen resolutions for these tools – in Linux using **System>Preferences>Monitors**, and Windows via **Start>Control Panel>Display**)

From **UBUNTU**, **ping** the **WINDOWS2003** VM and view the contents of the Ping packets.

L3.7 Basic Host Discovery can be performed using ICMP or ARP traffic, typically with tools such as **ping** and **arping**. This type of active network scanning is easy to detect using an Intrusion Detection System (IDS), such as Snort.

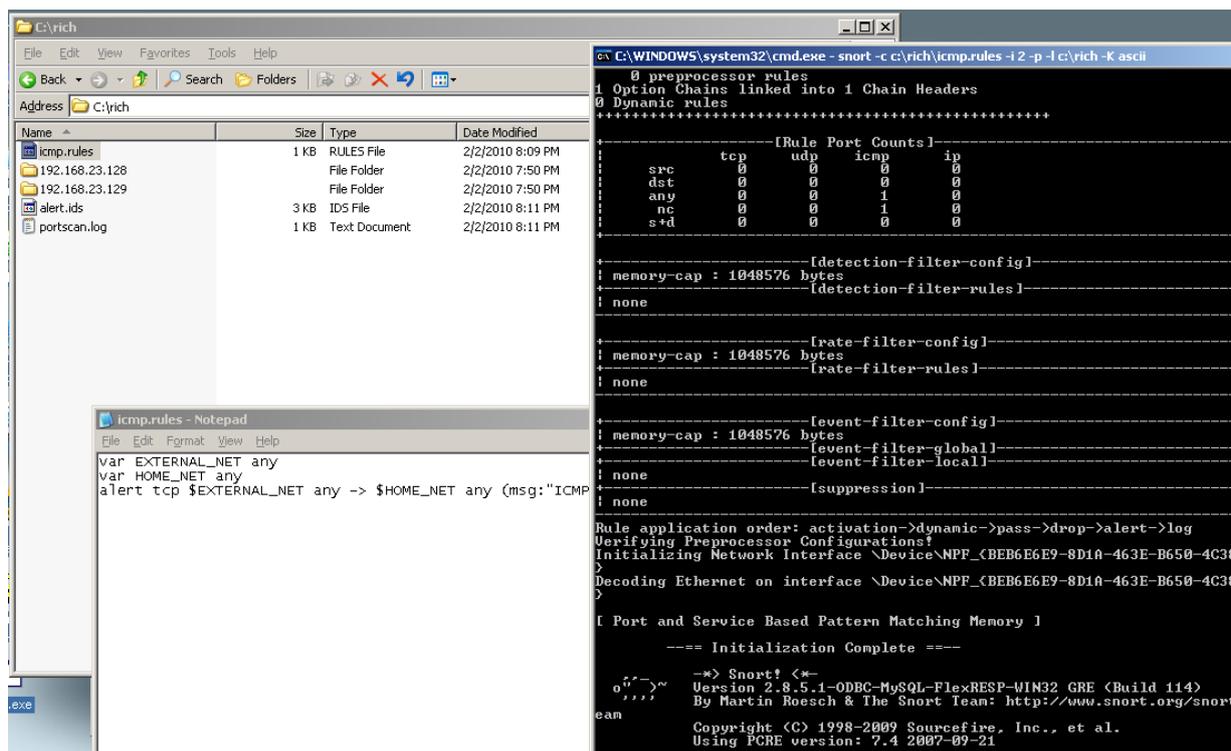
From **WINDOWS2003**, create a folder named **zzzzzzzz** (where **zzzzzzzz** is your matriculation number) and create a **snort detection rules file** in this folder named **icmp.rules**, and add the following snort variables, and detection rule:

```
var EXTERNAL_NET any
var HOME_NET any
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP ping"; sid:999)
```

and run Snort on WINDOWS2003 with:

```
snort -c c:\zzzzzzzz\icmp.rules -i 2 -p -l c:\zzzzzzzz -K ascii
```

Arrange your windows similar to shown below:



From UBUNTU, ping the WINDOWS2003 VM.

☞ Did Snort detect the pings from UBUNTU:

Then from UBUNTU, perform an ICMP Host Scan against the WINDOWS2003 VM, using nmap with `nmap -PE [WINDOWS2003 IP Address]`

☞ Did Snort detect the Host Scan from UBUNTU:

☞ To assist with this part of the lab and the following demo can be used:
http://buchananweb.co.uk/e_presentations/asfn_lab03_part01/asfn_lab03_part03.html

L3.8 Scanning specific hosts to find the services they are running is another common technique. This can be detected network auditing systems, by collecting traffic streams together and analysing them for scanning packets.

From WINDOWS2003, create a new IDS detection rules file call **portscan.rules** which will detect network scanning traffic, and add:

```
var EXTERNAL_NET any
var HOME_NET any
preprocessor sfportscan: proto { all } scan_type { all } sense_level { high }
logfile { portscan.log }
```

Run Snort with the detection portscan rules on WINDOWS2003 with:

```
snort -c c:\zzzzzzzz\portscan.rules -i 2 -p -l c:\zzzzzzzz -K ascii
```

and from UBUNTU, perform a **Port Scan** on WINDOWS2003 using:

```
nmap IPAddress.
```

☞ Did **Snort** detect the port scan:

☞ What type of port scan has been performed (which protocol is being used):

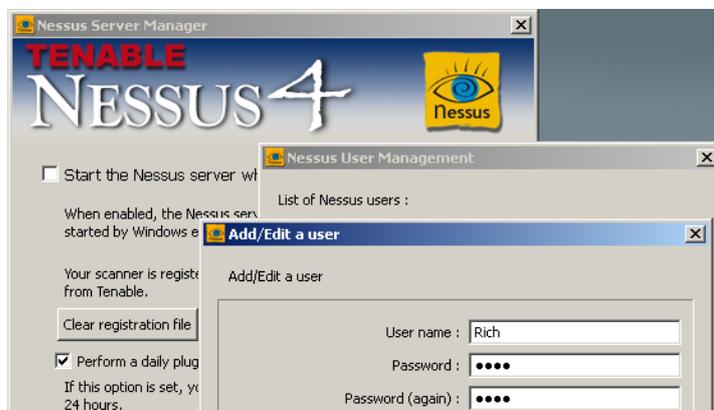
3.2.4 Vulnerability Scanning with the Nessus Scanner

Automated Vulnerability Scanners such as **Nessus** searches for known vulnerabilities on target systems, and report them, sometimes suggesting fixes for problems.

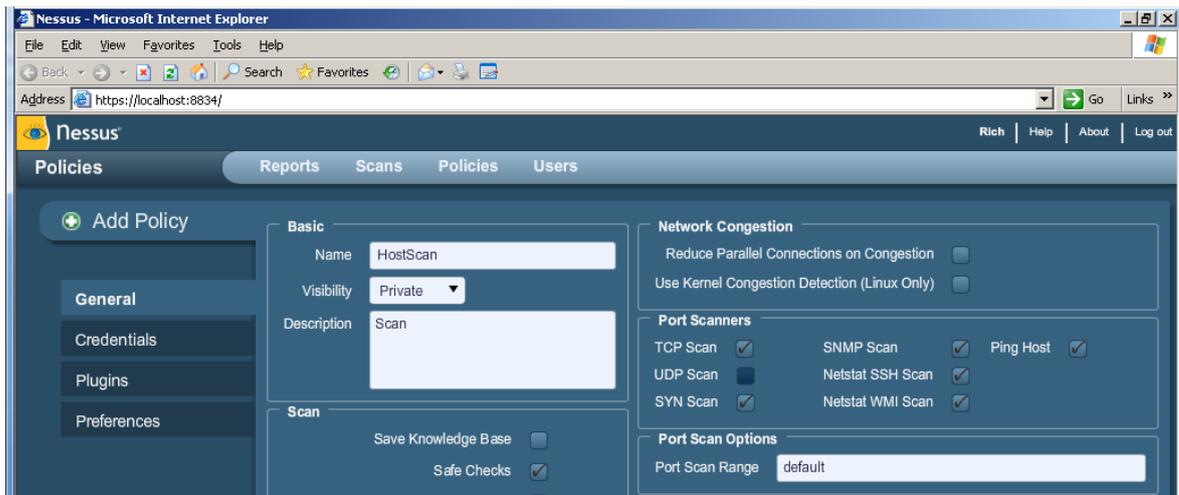
Nessus Demo:

http://buchananweb.co.uk/adv_security_and_network_forensics/nessus/nessus.htm

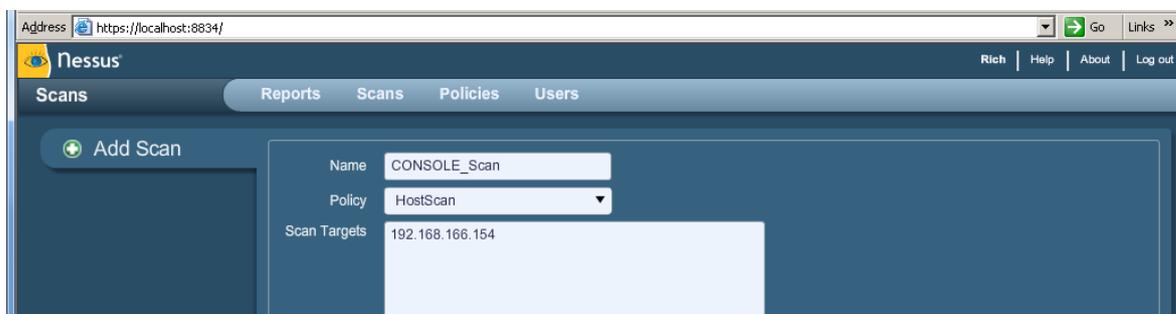
L3.9 From WINDOWS2003, run the **Nessus** vulnerability scanner **Server Manager** application, and create a new user account with the **Manage Users** button.



Create a new **Policy** called **PortScan**. Select some Port Scanning items, such as for TCP Scan and Syn Scan, as shown below.



Create a new **Scan** using the Policy you created.



Run the new scan against UBUNTU to discover the services which are running and to highlight any vulnerabilities with the services. Double click the scan as it runs, to see the progress.



Assess the services/vulnerabilities it reports them. Each service it finds can be double clicked to get more details.

☞ What are the Services found, including any type and version information reported by Nessus?

Note: Nessus reports on possible vulnerabilities and possible fixes automatically.

3.2.5 Password Cracking with Hydra

THC Hydra is a network service password cracking tool which can be used against the authentication of many network services, such as FTP and HTTP.

More information on **THC Hydra**, can be found at:
<http://www.thc.org/thc-hydra>

Video Demo:

http://buchananweb.co.uk/e_presentations/asfn_lab03_part04/asfn_lab03_part04.html

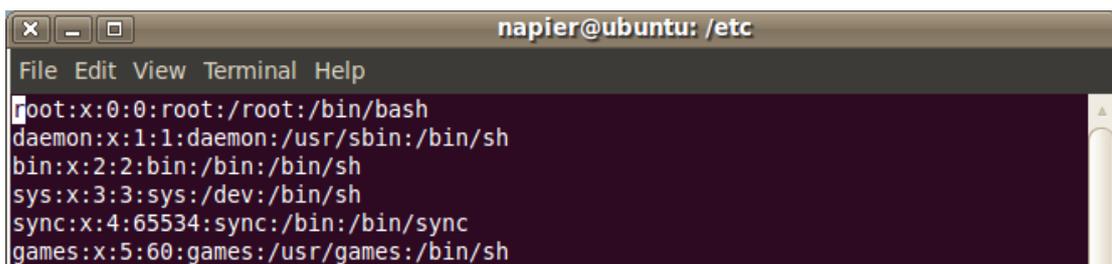
NOTE: Hydra should only be used on the lab private network. Do not run on any systems on the Internet.

Linux User/Passwords

The Linux `/etc/passwd` file is used to store information about users, such as Password, User ID, Description, and the users Home Directory. Each user has a record in the file (a line in the file) with fields holding details on the user, separated by semi-colons.

Edit the `/etc/passwd` file using the vi editor:

```
vi /etc/passwd
```



```
napier@ubuntu: /etc
File Edit View Terminal Help
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
```

- What is the first user in the file?
- What is the last user in the file? (CTRL+F and CTRL+B can be used to page forward and back)
- The second field is the users password ... what value is this for all users?

`:q` can be used to exit the vi editor

Most Linux systems have an `x` in place of a password, and the actual passwords are hashed and stored in the `/etc/shadow` file, to provide an extra layer of security.

List the contents of the `/etc/shadow` file using the `more` command (sudo is needed as the shadow file needs root permissions to read):

```
sudo more /etc/shadow
```

L3.10 On UBUNTU create a new ftp user, using the `useradd` command, with your name, such as `fred` in the following example (check by viewing the `/etc/passwd` file):

```
cd /home
mkdir FTP-rich
sudo useradd richftp -d /home/FTP-rich -s /bin/false
```

Set the password to **password** for your new ftp user.

```
sudo passwd richftp
```

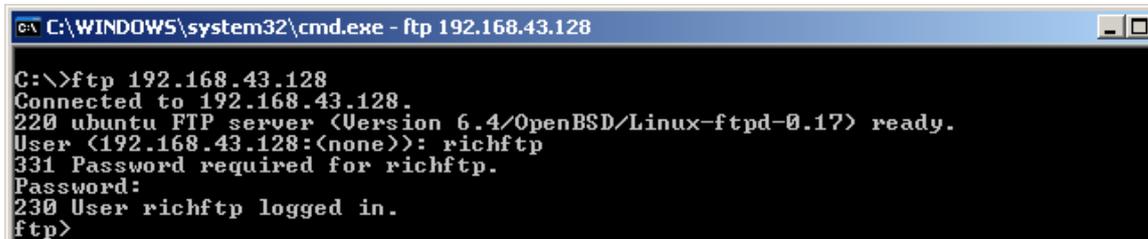
List the contents of the `/etc/passwd` file using the **more** command:

```
more /etc/passwd
```

☞ Has the new user been successfully created?

YES/NO

Next try go to WINDOWS2003 and log into the FTP server on UBUNTU with the username and password that you have created.



```
C:\WINDOWS\system32\cmd.exe - ftp 192.168.43.128
C:\>ftp 192.168.43.128
Connected to 192.168.43.128.
220 ubuntu FTP server (Version 6.4/OpenBSD/Linux-ftp-0.17) ready.
User (192.168.43.128:(none)): richftp
331 Password required for richftp.
Password:
230 User richftp logged in.
ftp>
```

☞ Can you log in with the new ftp user successfully?

YES/NO

From WINDOWS2003, go to the `C:\hydra` directory and add the new ftp user to the `login.txt` file using an editor such as notepad.

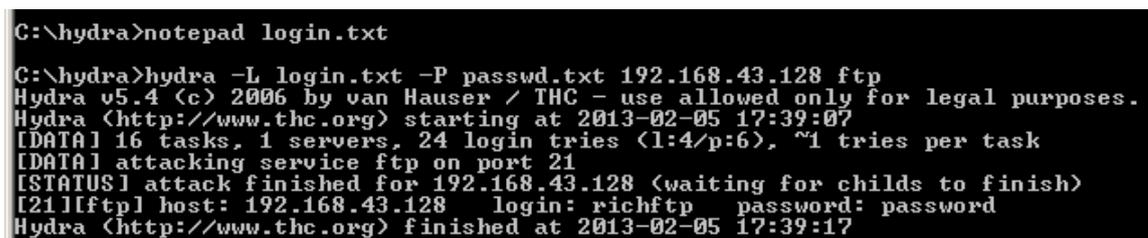
Try to crack the ftp password using **hydra**, the passwords file `passwd.txt` such as:

```
C:\hydra> hydra -L login.txt -P passwd.txt [UBUNTU IP ADDRESS] ftp
```

☞ Did hydra successfully crack the password

YES/NO

☞ What type of password cracking attack is this: dictionary/bruteforce?



```
C:\hydra>notepad login.txt
C:\hydra>hydra -L login.txt -P passwd.txt 192.168.43.128 ftp
Hydra v5.4 (c) 2006 by van Hauser / THC - use allowed only for legal purposes.
Hydra (http://www.thc.org) starting at 2013-02-05 17:39:07
[DATA] 16 tasks, 1 servers, 24 login tries (1:4/p:6), ~1 tries per task
[DATA] attacking service ftp on port 21
[STATUS] attack finished for 192.168.43.128 (waiting for childs to finish)
[21][ftp] host: 192.168.43.128 login: richftp password: password
Hydra (http://www.thc.org) finished at 2013-02-05 17:39:17
```

From UBUNTU, set the passwd for the ftpuser to something like richpass, and from WINDOWS2003 try to crack it using hydra again.

- | | |
|---|--------|
| ☞ Did hydra successfully crack the password | YES/NO |
| ☞ What would need to be improved to crack the password? | |

L3.11 Go UBUNTU, and run Wireshark, and rescan with Hydra, and capture the trace. Now find the successful login from the trace.

- | |
|--|
| ☞ Which is the message displayed on Wireshark: |
|--|

L3.12 If using the cloud, from a neighbours host, get them to add a new user (with an easy to guess username and password), and download a dictionary file and place into login.txt and password.txt, and see if you can determine the password.

L3.13 Can you create a rule which detects the Hydra footprint?

Possible Snort rule:

3.2.6 Database SQL Injection

SQL Injection Demo:

http://buchananweb.co.uk/adv_security_and_network_forensics/cross_script/cross_script.htm

L3.14 Using the WINDOWS2003 virtual machine (User name: Administrator, Password: napier), run **Visual Web Developer Express 2008** (register with your live email account if necessary), and select **File->Open Web Site**, and select **c:\inetpub\wwwroot** to open the default website.

L3.15 Open the **Database Explorer** Panel (**View->Database Explorer**), click the **Connect to Database** button, and setup a Connection to the Sample DB as shown in Figure L3.1.

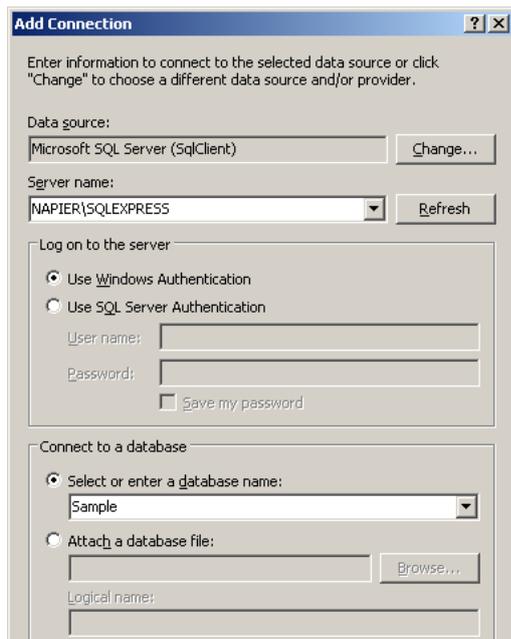


Figure L3.1 Database connection

L3.16 Create a new **databasesample.aspx** Web page, and add a GridView component to load db data into. Double click on the form, and then add the following highlighted code to the web page:

```
using System.Web.UI.WebControls;
```

```
using System.Data.Sql;
```

```
using System.Data.SqlClient;
```

```
using System.Data.SqlTypes;
```

```
namespace WebApplication2
```

```
{
```

```
    public partial class _Default : System.Web.UI.Page
```

```
    {
```

```
        protected void Page_Load(object sender, EventArgs e)
```

```
        {
```

```
            SqlCommand s = null;
```

```
            string param = Request.QueryString["test"];
```

```
            MySqlConnection = createConn("Sample");
```

```
            MySqlConnection.Open();
```

```
            s = new SqlCommand("SELECT * FROM db1", MySqlConnection);
```

```
            if (param != null) s = new SqlCommand(param, MySqlConnection);
```

```
            SqlDataReader myDataReader = s.ExecuteReader();
```

```
            GridView1.DataSource = myDataReader;
```

```
            GridView1.DataBind();
```

```
            closeConn();
```

```
        }
```

Next add the following code to the class, just below the Page_Load event:

```

public SqlConnection mySqlConnection;
public SqlCommand mySqlCommand;
public SqlDataReader mySqlDataReader;

private void closeConn()
{
    if (mySqlConnection != null)
    {
        if (mySqlConnection.State == ConnectionState.Open)
        {
            mySqlConnection.Close();
        }
        mySqlConnection.Dispose();
    }
}

private SqlConnection createConn(string database)
{
    string mySqlConnectionString =
@"Data Source=NAPIER\SQLEXPRESS;Initial Catalog=Sample;
Integrated Security=True";

    if (mySqlConnection == null) {
        mySqlConnection = new SqlConnection(mySqlConnectionString); };

    return mySqlConnection;
}

```

L3.17 Set `databasesample.aspx` as the default **Start Page**, and test the web page, using **Debug->Start Debugging (F5)**.

☞ What are some of the contents of the table?

☞ How many rows are in the table?

L3.18 Next comment out the line:

```
// s = new SqlCommand("SELECT * FROM db1", mySqlConnection);
```

and add the line:

```
s = new SqlCommand("INSERT INTO db1 VALUES
('Bert', 'Smith4', '25', '25', '35')", mySqlConnection);
```

and run the code. After this replace the original line, and rerun the code.

☞ What are the contents of the table, and how many rows in the table?

☞ Has a new line been added?

L3.19 Next comment out the line:

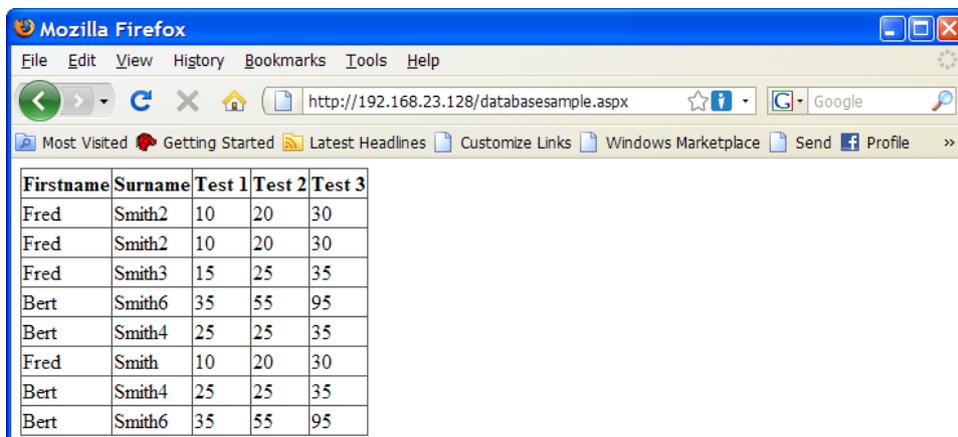
```
// s = new SqlCommand("SELECT * FROM db1", mySqlConnection);  
and add the line:  
s = new SqlCommand("SELECT * FROM db1 WHERE Surname LIKE 'Smith%',  
mySqlConnection);
```

☞ What are the contents of the table, and how many rows are displayed?

L3.20 Next from the Host computer (DESKTOP), access the Web Page on the WINDOWS2003 Web Server, via a browser with:

<http://192.168.n.n/databasesample.aspx>

☞ What are the contents of the table, and how many rows are displayed?



Firstname	Surname	Test 1	Test 2	Test 3
Fred	Smith2	10	20	30
Fred	Smith2	10	20	30
Fred	Smith3	15	25	35
Bert	Smith6	35	55	95
Bert	Smith4	25	25	35
Fred	Smith	10	20	30
Bert	Smith4	25	25	35
Bert	Smith6	35	55	95

Figure 1 - Accessing the WINDOWS2003 Web Page

L3.17 Next from the Host computer (DESKTOP), create a SQL Injection attack against the WINDOWS2003 Web Server, via the URL, with:

http://192.168.n.n/databasesample.aspx?test=SELECT%20*%20FROM%20db1

☞ What are the contents of the table, and how many rows are displayed?

L3.21 Next from the Host computer (DESKTOP), create a SQL Injection attack against the WINDOWS2003 Web Server with:

```
http://192.168.n.n/databasesample.aspx?test=INSERT%20INTO%20db1%20VALUE
S%20('Bert','Smith7','35','55','95')
```

followed by:

```
http://192.168.75.132/databasesample.aspx?test=SELECT%20*%20FROM%20db1
```

☞ What are the contents of the table, and how many rows are displayed?

☞ Has the SQL Injection attack been successful in adding a new row?

L3.22 Create an **SQL statement** from the aspx code, to calculate the average mark for Test 1, such as:

```
s = new SqlCommand("SELECT avg([Test 1]) FROM db1", mySqlConnection);
```

Test on the local Web server, and then use an SQL injection from a URL. Repeat for the minimum and maximum mark for Test 1.

L3.23 Create a **SQL injection attack**, to change Ian Archibalds mark to 100%.

L3.24 Modify the **aspx code** so that it detects a SQL injection attack, and identifies the SQL statement used.

3.3 Toolkit Development 3 – Integrate Snort/Nmap

☞ Video of part 3 of the toolkit software development:

http://buchananweb.co.uk/adv_security_and_network_forensics/toolkit02/toolkit02.htm

The objective of this series of labs is to build an integrated toolkit. Open up:

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

and extract to a local folder. Next open up toolkit.sln, and double click on client.cs.

☞ For more on **Snort** see: <http://www.snort.org/>

For more on **nmap** see: <http://nmap.org/>

L3.25 Select the **Snort** tab, and double click on the “Show Interf” button and add the following code:

```
runProgram3("snort", "-W");
```

on View IDS add:

```
Process.Start("notepad.exe", "log\\alert.ids");
```

on the View button:

```
Process.Start("notepad.exe", tbSnortFile.Text);
```

on View ARP:

```
Process.Start("wordpad.exe", "log\\ARP.");
```

on Delete alert.ids:

```
try {
    File.Delete("log\\alert.ids");
    File.Delete("log\\ARP");
}
catch (Exception ex)
{ }
```

on the ComboBox (cbInterfacesSnort) and add the code:

```
stopProcess("snort");
if (processCaller3!=null) processCaller3.Cancel();

processCaller3 = null;

int ind = cbInterfacesSnort.SelectedIndex + 1;
string args = " -p -K ascii -N ";
args += " -i " + ind;
if (this.cbVerboseSnort.Checked) args += " -v ";
if (tbSnortFile.Text.Length > 0) args += " -c " + tbSnortFile.Text;

runProgram3("snort.exe", args);
timer2.Enabled = true;
```

L3.26 Download the **nmap** network scanner, and integrate it in with the toolkit (add a box for an IP address to scan), and an options box.