Lab 1

Setting up VMs for Hacking Labs

1 Introduction

In a typical hacking attack that occurs across a network, the attacker will launch the attack from a computer that is physically distinct from the server or machine under attack. The computer used to attack may be a remote machine on the Internet or may be located within the internal LAN that the target machine is connected to (Figure 1).

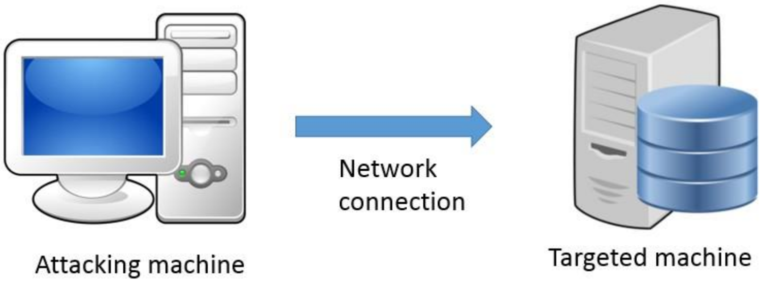


Figure 1

To simulate such an attack in our lab, we can designate two separate computers as the attacking and targeted computer respectively. However, this is problematic because of two reasons:

1. Since the lab will be nearly fully occupied, each student will have to take turns to be the attacker and targeted machine respectively as it may not be possible in some scenarios to attack and be attacked simultaneously. This slows down the progression of the labs.

2. Some types of attacks, such as a denial of service attack, will flood the network connection with packets. If too many machines participate in the attack at once, the LAN in the lab may not have sufficient bandwidth to accommodate all the packets necessary for a successful attack to occur.

To resolve these issues, a common approach employed in hacking labs is to make the designated target computer a VM running within a host OS. In the simplest case, the host machine can be the attacking machine with a virtual network connection providing the transit for the attack to occur (Figure 2)

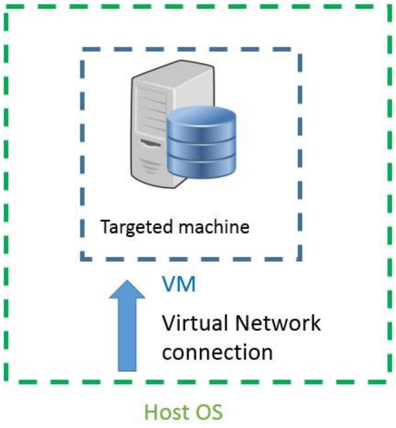


Figure 2

Further extending on this basic setup, another VM can be generated within the host OS to play the role of the attacking machine. This can be useful if the host OS does not provide the required functionalities for an attack to occur; in which case the required OS for the attacking machine becomes the guest OS within the second VM. (Figure 3).

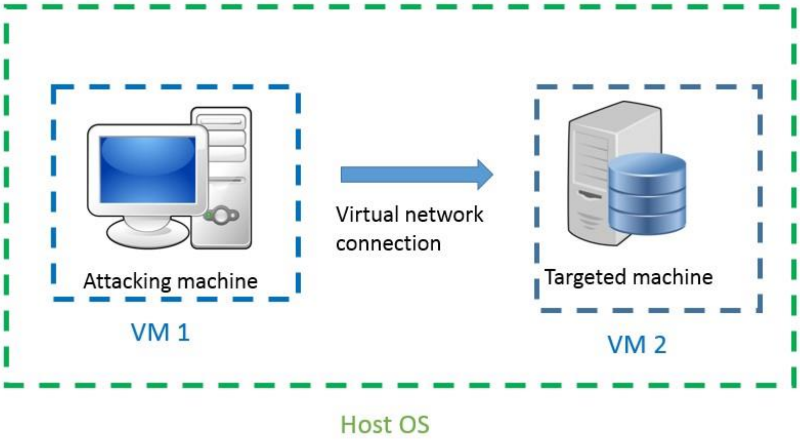


Figure 3

We can further opt to set up 3 or more VMs within a single host to emulate a hacking attack that involves multiple attacking machines, for e.g. in a denial of service attack. The number of VMs that can be hosted in this way will be very much dependent on the hardware resources (memory, hard disk space, CPU capacity) of the hosting machine.

The advantages of using VMs to emulate a network hacking attack are:

1. The VM is a self-contained independent module which exists as a collection of files representing a virtual hard disk. Any damage or corruption to the guest OS within the VM will not in any way affect the host OS, its file system or the underlying machine.

2. Setting up and deploying an OS in a VM is typically faster than installing it on a physical hard disk, particularly when dual booting is involved. Therefore, a corrupted or damaged VM can be discarded and a new one set up quickly.

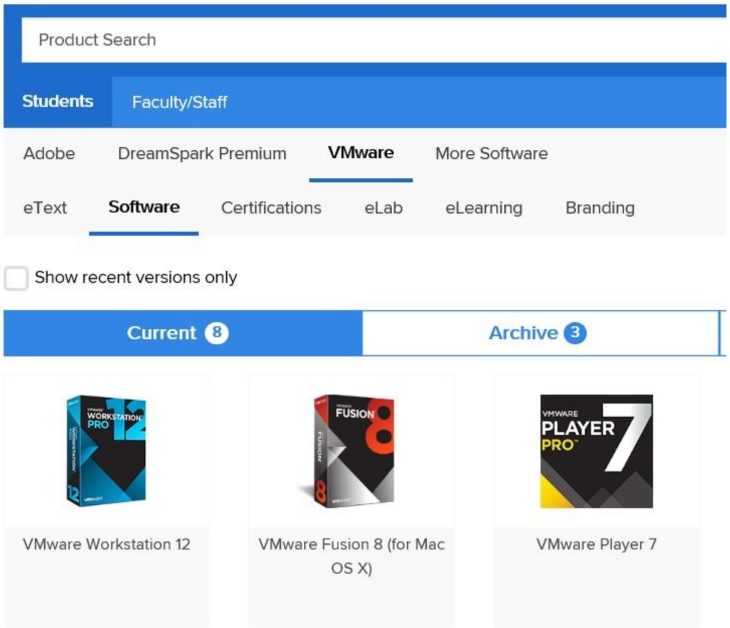
3. Since both the attacking and targeted machines are emulated on the same machine, the student can configure both of them more easily in order to control and understand the attack mechanism more easily

4. Attacks that involve flooding the network with packets can be performed on a virtual network involving one or more VMs without serious disruption to the LAN that the host machine is connected to.

The primary disadvantage of course is that a VM only provides an emulation of the behavior of a physical machine and network. A real life attack will occur over a network where traffic load varies greatly and with some other network security controls present that are not so easily emulated within a virtual environment. However, using VMs provides a very useful start for learning hacking methodology within a contained environment.

2 Obtaining Virtualization Software

The most well known virtualization applications for the Windows desktop platform is VMWare Workstation Pro, which has a commercial as well as trial versions. Oracle VirtualBox is another well known and widely used open source virtualization platform. VMWare Workstation is available for free under the Microsoft Dreamspark programme which our faculty participates in. You can get a Dreamspark account set up for by one of the level 6 lab staff (Bob or Teo), after which you can download related Microsoft and VMWare applications along with an activating license key.



They can also be downloaded from: https://www.vmware.com/products/workstation/
Note that VMWare Workstation requires a 64-bit processor (minimum Intel Core 2 Duo; recommended Intel I5 or higher) as well as a 64 bit host OS.
Oracle Virtualbox is available at: https://www.virtualbox.org/
and is installable on both 32 as well as 64 bit OS.

3 Key tasks in VMWare Workstation

We briefly cover here the main tasks that you will need to be acquainted with while working with VMWare Workstation to generate the various guest OS required for the hacking labs. You should have access to the user manual (`workstation-pro-12-user-guide.pdf`) for this purpose.

3.1 Installing a new guest OS

User manual: pg. 41 – 46, 51 – 54

You need to ensure you have the relevant ISOs for the guest OS (Windows / Linux) of the VM that need to be installed beforehand. You may need to customize the directory location to hold the virtual hard disk (`*.vmdk` files), which can range from 1GB to 1TB, depending on what you intend to store on it. Other relevant parameters include network adapter settings, memory and CPU allocation. You should fine tune these parameters to optimize the performance of the VMs that you intend to run on your host OS in accordance to the physical resources available on the host machine.

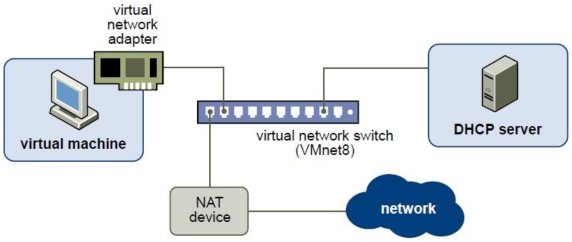
3.2 Configuring network connections

User manual: pg. 177 – 181

The networking configuration of the VMs representing the attacking and targeted machine should be set either to NAT mode or bridged mode for the labs. Either mode should allow the guest OS access to the Internet if the host machine is already set up for Internet access through a LAN (in the labs) or broadband modem (at home).

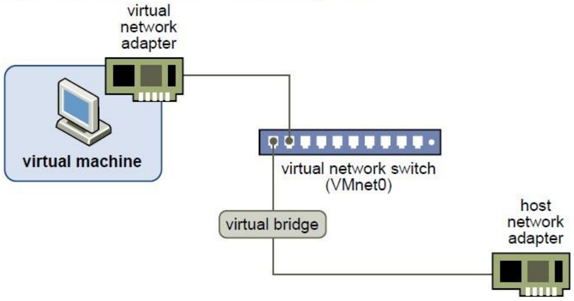
With NAT, a virtual machine does not have its own IP address on the external network. Instead, a separate virtual private network is set up on the host system. In the default configuration, the virtual machine obtains its IP address on this private network from the virtual DHCP server created by the hypervisor. The virtual machine shares the network identity of the host machine, and is therefore not visible to other machines on the physical network (LAN) that the host machine is attached to. NAT works by translating the IP addresses of virtual machines in the private network to the IP address that host machine has on the physical network. When a virtual machine sends a request to access a resource on the physical network, it appears to the resource as if the request is coming from the host machine instead.

Figure 9-2. NAT Configuration



In bridged mode, the virtual network adapter in the VM connects to a physical network adapter in the host machine. The host network adapter enables the VM to connect to the same physical network (LAN) that the host machine is attached to. The VM therefore has a unique identity with its own IP address on the physical network, and is considered separate and unrelated to the host machine. It can access other machines on the network, which can also view and contact it as if it were a physical computer on the network. The IP allocated for the VM will be within the valid network address range of the LAN subnet that the host machine is connected to. This IP can be automatically allocated (if there is a DHCP server within that LAN) or it can be manually allocated (through the network settings in the guest OS of the VM).

Figure 9-1. Bridged Networking Configuration



3.3 Using Shared Folders

User manual: Pg. 84 – 87

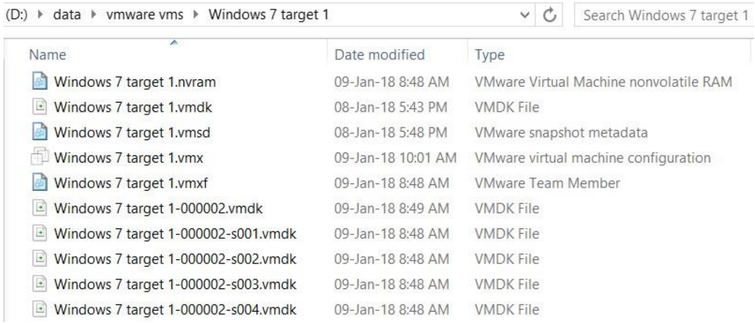
Shared folders allow you to transfer files between the host OS and the guest OS easily. This may be useful when you wish to install specific hacker tools on a guest OS that is to be used an attacking machine. For a Windows 7 or 8 guest OS, it will be easier to map the shared folder as a network drive in the guest OS. For Windows Server 2008 or 2012 guest OS, a shared network folder should be created on the host OS which should be visible within the guest OS if the virtual network connection is configured properly. The shared network folder approach should also work with a Windows 7 or 8 guest OS.

Alternatively, you can drag and drop folders from the file system of the host OS into that of the guest OS. The VMTools facility of VMWare Workstation, which is usually installed with the creation of the VM, provides support for this.

3.4 Relocating Virtual Machines

User manual: pg. 129 – 131

The lab machines are configured to remove all files and software installations on them when they are rebooted for security reasons. This means that the files that you have saved on a VM or the software that you have installed will be lost once you reboot the host OS. You can copy the folder containing the files that constitute the virtual hard disk for the VM to a portable storage device after you are done with a lab session if you wish to retain these changes.

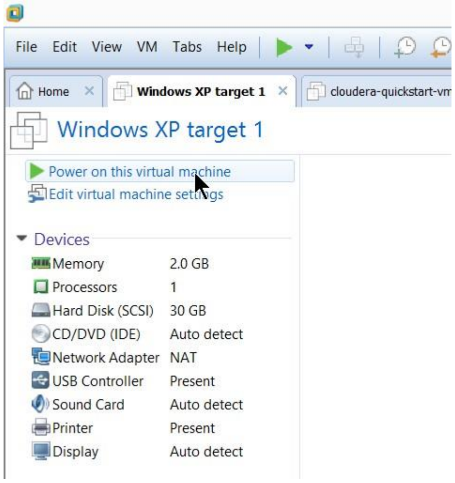

(D:) ▸ data ▸ vmware vms ▸ Windows 7 target 1
Search Windows 7 target 1

Name	Date modified	Type
Windows 7 target 1.nvram	09-Jan-18 8:48 AM	VMware Virtual Machine nonvolatile RAM
Windows 7 target 1.vmdk	08-Jan-18 5:43 PM	VMDK File
Windows 7 target 1.vmsd	08-Jan-18 5:48 PM	VMware snapshot metadata
Windows 7 target 1.vmx	09-Jan-18 10:01 AM	VMware virtual machine configuration
Windows 7 target 1.vmxf	09-Jan-18 8:48 AM	VMware Team Member
Windows 7 target 1-000002.vmdk	09-Jan-18 8:49 AM	VMDK File
Windows 7 target 1-000002-s001.vmdk	09-Jan-18 8:48 AM	VMDK File
Windows 7 target 1-000002-s002.vmdk	09-Jan-18 8:48 AM	VMDK File
Windows 7 target 1-000002-s003.vmdk	09-Jan-18 8:48 AM	VMDK File
Windows 7 target 1-000002-s004.vmdk	09-Jan-18 8:48 AM	VMDK File

4 Running hacking labs on the uni machines

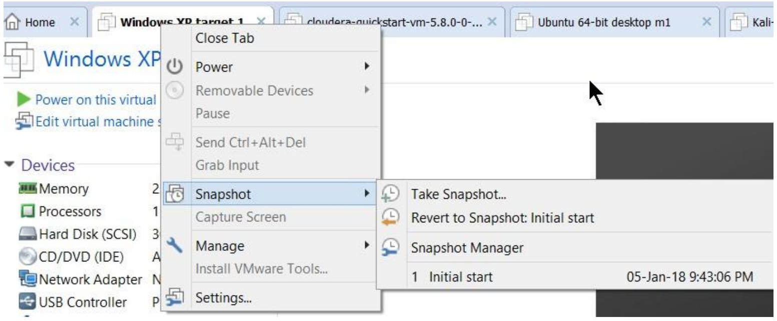
4.1 Restarting the VMs from a snapshot

Typically you will start a VM by clicking on its tab in main VMWare Workstation window, and then selecting `Power on this Virtual machine`.


File Edit View VM Tabs Help
Home
Windows XP target 1
cloudera-quickstart-vm
Windows XP target 1
Power on this virtual machine
Edit virtual machine settings
Devices

Memory	2.0 GB
Processors	1
Hard Disk (SCSI)	30 GB
CD/DVD (IDE)	Auto detect
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

However, occasionally your VM may not start properly or alternatively, you may have made certain changes to the host OS in the VM during the course of a hacking exercises that result in it malfunctioning. In that case, you can restart the VM using a snapshot. Right click on the tab of the VM concerned, select `Snapshot` and select `Revert to Snapshot: Initial Start`.



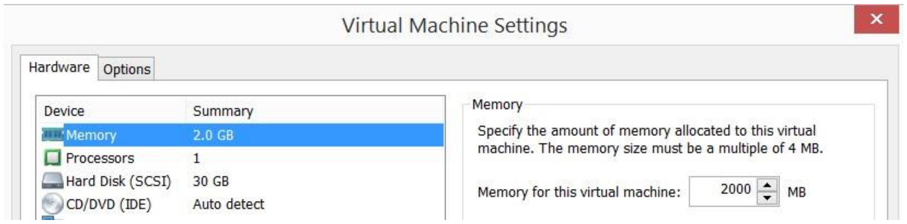
A snapshot is basically a saved state of the guest OS at a particular point in time during its execution. It is primarily created so that we can restore the saved state in the event that unwanted changes have occurred to the guest OS that are difficult or impossible to undo. When the VMs for this lab was created, I created an initial snapshot for all the VMs for this purpose. You can also create your own snapshots during the progression of a lab exercise in order to maintain your progress. This way if the VM hangs and crashes, you can restore from the most recent snapshot rather than restarting the entire lab exercise.

4.2 Ensuring sufficient system resources for the VMs

All VMs are allocated specific system resources from the host machine, which you can view by clicking on the tab of the VM concerned in the main VMWare Workstation window. The two most important resources to keep track of are memory and processors. You need to ensure that the combined memory allocated to all the running VMs still leaves sufficient space in the RAM of the host machine to run the host OS efficiently.

For e.g. if you allocate 3GB each to the 3 VMs and you run all of them at the same time, your VMs can effectively consume up to maximum total of 9 GB. If the host machine has 12 GB of RAM, then there is adequate space left (3 GB) to run the host OS efficiently. On the other hand, a host machine with 8 GB of RAM (such as the lab machines in KB605) would slow down to a crawl running 3 VMs at 3 GBs each. In that case, we would need to change the memory allocations for some or all of the VMs involved. Alternatively, we could ensure that we only run any 2 of the 3 VMs at the same time. Typically, we would try to leave at least 2 GB of RAM for running the host OS.

You can change the allocations for memory (and other hardware resources) by selecting Edit Virtual Machine Settings for a given VM tab and then changing the values in the Virtual Machine Settings dialog box.



Device	Summary
Memory	2.0 GB
Processors	1
Hard Disk (SCSI)	30 GB
CD/DVD (IDE)	Auto detect

Memory

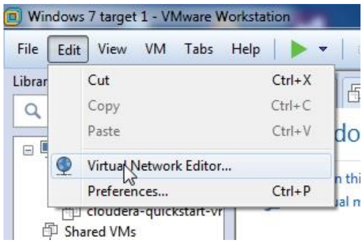
Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

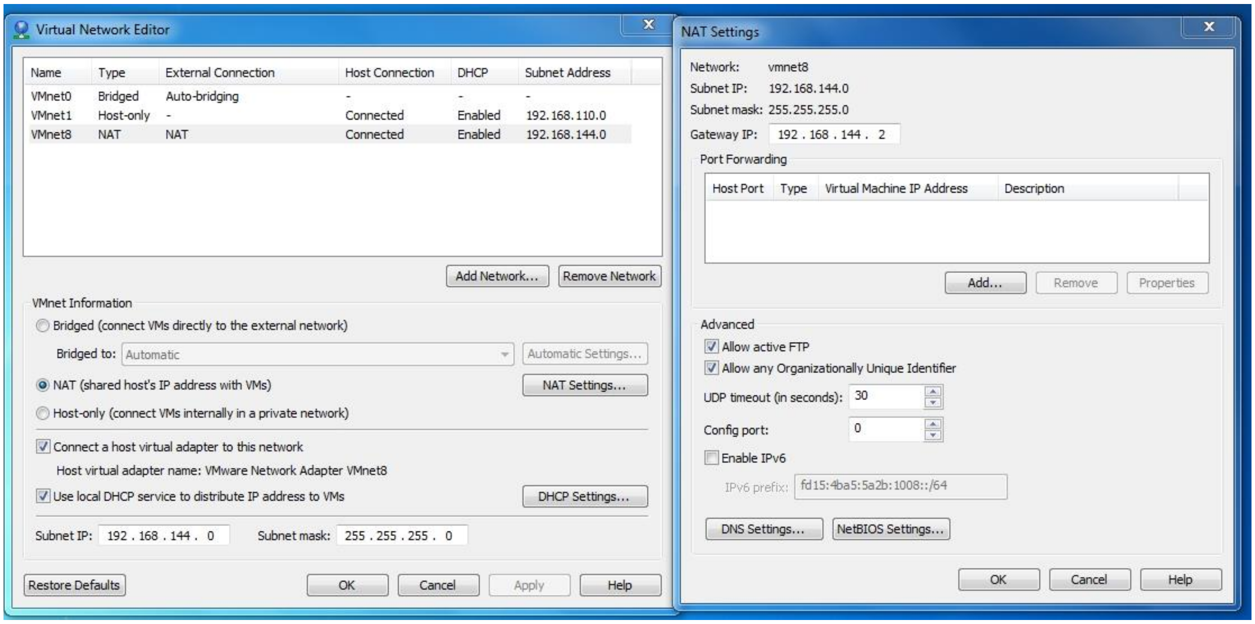
Memory for this virtual machine: 2000 MB

4.3 NAT settings for the VMs

For the labs to run properly, the attacking machine(s) must be able to access the targeted machine(s) across a real or virtual network connection. If either (or both) the attacking and targeted machines are set up as VMs, configuring the virtual network adapter into either the NAT or bridged mode for the VMs will fulfill this requirement. Note that if you are setting up both the attacking and targeted machines as VMs, then both must be configured with the same networking mode (NAT or bridged).

For most of the labs, we will be running the VMs in NAT mode. We can determine the subnet address and gateway for the NAT virtual network for the particular version of VMWare Workstation by selecting `Edit -> Virtual Network Editor -> NAT Settings`.





Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet0	Bridged	Auto-bridging	-	-	-
VMnet1	Host-only	-	Connected	Enabled	192.168.110.0
VMnet8	NAT	NAT	Connected	Enabled	192.168.144.0

VMnet Information

○ Bridged (connect VMs directly to the external network)

Bridged to: Automatic

● NAT (shared host's IP address with VMs)

○ Host-only (connect VMs internally in a private network)

☑ Connect a host virtual adapter to this network

Host virtual adapter name: VMware Network Adapter VMnet8

☑ Use local DHCP service to distribute IP address to VMs

Subnet IP: 192 . 168 . 144 . 0 Subnet mask: 255 . 255 . 255 . 0

Network: vmnet8

Subnet IP: 192.168.144.0

Subnet mask: 255.255.255.0

Gateway IP: 192 . 168 . 144 . 2

Port Forwarding

Host Port	Type	Virtual Machine IP Address	Description

Advanced

☑ Allow active FTP

☑ Allow any Organizationally Unique Identifier

UDP timeout (in seconds): 30

Config port: 0

☐ Enable IPv6

IPv6 prefix: fd15:4ba5:5a2b:1008::/64

For this particular VMWare Workstation, we can see the network address for the NAT is 192.168.144. Therefore, any VM started in the NAT mode can be assigned an address between 192.168.144.1 – 192.168.144.254 (255 is the broadcast address for the subnet), either statically or via the virtual DHCP server.

4.4 Setting static IP addresses for VMs in the NAT

We will be mainly working with 4 VMs in this lab: Windows 7 VM, Windows XP VM, Kali Linux VM and Metasploitable VM. It is easier to use a static IP address for all the VMs as the virtual DHCP server may not work every time.

The admin account and passwords for these 4 VMs are:

`Windows XP VM`
Admin account: **winxpadmin**

Password: **monkey**

`Windows 7 VM`
Admin account: **win7admin**
Password: **donkey**

`Kali Linux VM`
Admin account: **root**
Password: **toor**

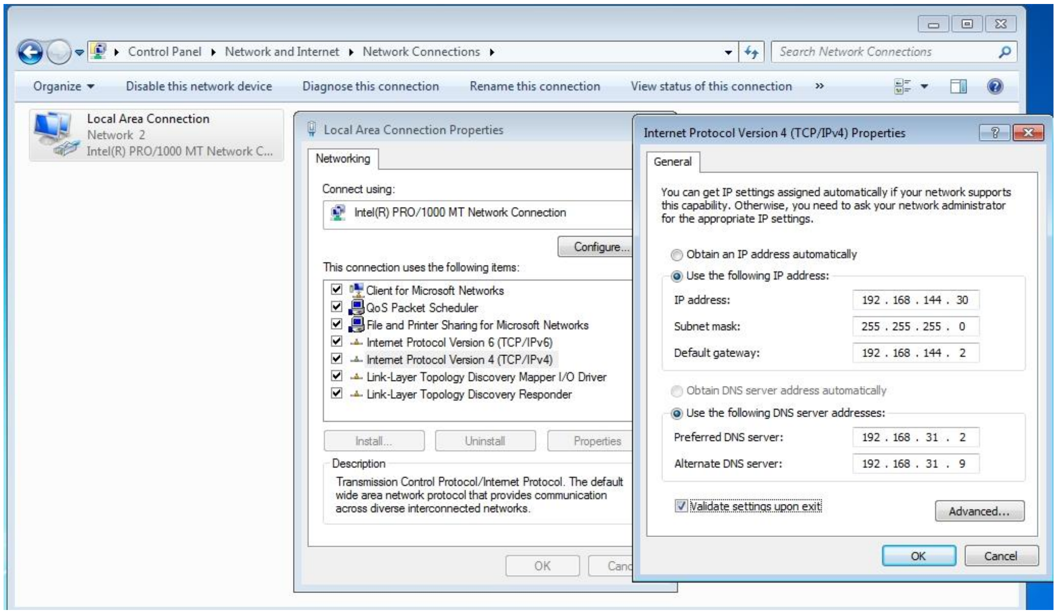
`Metasploitable VM`
Admin account: **msfadmin**
Password: **msfadmin**

To set the static IP address for the Windows 7 VM, login to it with the admin account, then select `Control Panel -> Network and Internet -> Network and Sharing Center`. Select `Change Adapter Settings`, then right click on the `Local Area Connection` icon and then select `Properties`. Then select `Internet Protocol Version 4 (TCP/IPv4)` and then select `Properties`. In the TCP/IPv4 properties dialog box that appears, set the values for the fields as follows:

`Windows 7 VM`

IP address: **129.168.144.30**
Subnet mask: **255.255.255.0**
Default gateway: **192.168.144.2**

Preferred DNS server: **192.168.31.2**
Alternate DNS server: **192.168.31.9**



Control Panel ▸ Network and Internet ▸ Network Connections ▸
Search Network Connections
Organize ▾
Disable this network device
Diagnose this connection
Rename this connection
View status of this connection
»
Local Area Connection
Network 2
Intel(R) PRO/1000 MT Network C...
Local Area Connection Properties
Networking
Configure...
Install...
Uninstall
Properties
OK
Canc
Internet Protocol Version 4 (TCP/IPv4) Properties
General
Advanced...
OK
Cancel

Connect using:
Intel(R) PRO/1000 MT Network Connection

This connection uses the following items:
☑ Client for Microsoft Networks
☑ QoS Packet Scheduler
☑ File and Printer Sharing for Microsoft Networks
☑ Internet Protocol Version 6 (TCP/IPv6)
☑ Internet Protocol Version 4 (TCP/IPv4)
☑ Link-Layer Topology Discovery Mapper I/O Driver
☑ Link-Layer Topology Discovery Responder

Description
Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:
IP address: 192 . 168 . 144 . 30
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 192 . 168 . 144 . 2

○ Obtain DNS server address automatically
● Use the following DNS server addresses:
Preferred DNS server: 192 . 168 . 31 . 2
Alternate DNS server: 192 . 168 . 31 . 9

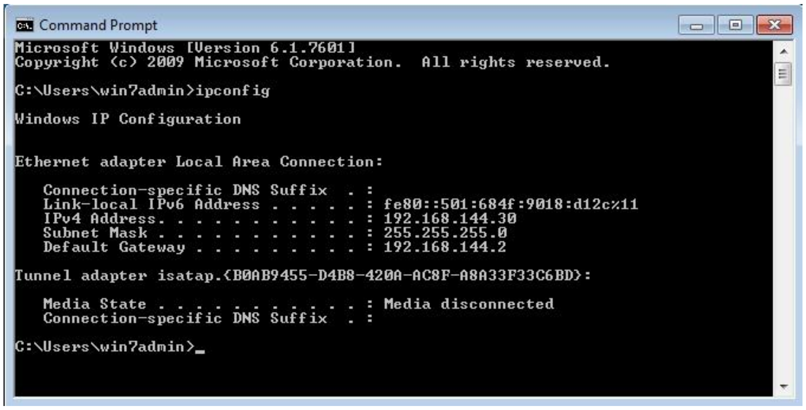
☑ Validate settings upon exit

When you are done click ok.

Once you have done this, open a command prompt in the Windows 7 VM and type

```
ipconfig
```

to verify the IP address has being set properly.

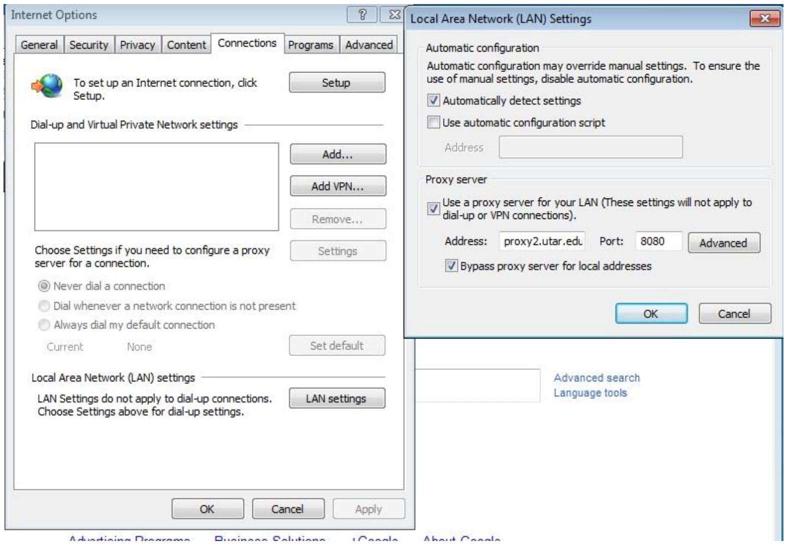


```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\win7admin>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::501:684f:9018:d12c%11
   IPv4 Address. . . . . . . . . . . : 192.168.144.30
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.144.2

Tunnel adapter isatap.{B0AB9455-D4B8-420A-AC8F-A8A33F33C6BD}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Users\win7admin>_
```

In order to connect to the Internet through the VM while using the university lab machines, we will need to configure the browser to go through the university proxy server. This step is not necessary if you are running the lab exercises on your home PC without any proxy server configured.

Open Internet Explorer in the VM, select the `Connections` tab, then select `LAN Settings`, and then enter the values for the fields below:

Address: **proxy2.utar.edu.my**
Port: **8080**



To set up an Internet connection, click Setup.

Dial-up and Virtual Private Network settings

Choose Settings if you need to configure a proxy server for a connection.

● Never dial a connection
○ Dial whenever a network connection is not present
○ Always dial my default connection

Current None

Local Area Network (LAN) settings

LAN Settings do not apply to dial-up connections. Choose Settings above for dial-up settings.

Automatic configuration

Automatic configuration may override manual settings. To ensure the use of manual settings, disable automatic configuration.

☑ Automatically detect settings
☐ Use automatic configuration script

Address

Proxy server

☑ Use a proxy server for your LAN (These settings will not apply to dial-up or VPN connections).

Address: proxy2.utar.edu Port: 8080

☑ Bypass proxy server for local addresses

Open Internet Explorer and browse to www.google.com to verify that you have Internet connectivity from the VM through the university proxy server.

Repeat the same procedure to set a static IP address for the Windows XP VM (logging in with the admin account as well), but with a different IP address.

`Windows XP VM`

IP address: **129.168.144.20**
Subnet mask: **255.255.255.0**
Default gateway: **192.168.144.2**

Preferred DNS server: **192.168.31.2**
Alternate DNS server: **192.168.31.9**

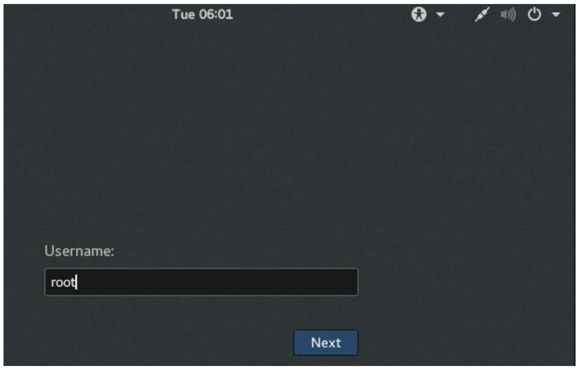
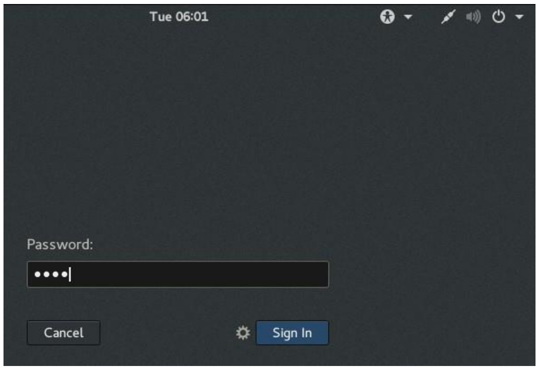
Again, open a command prompt in the Windows 7 VM and type

```
ipconfig
```

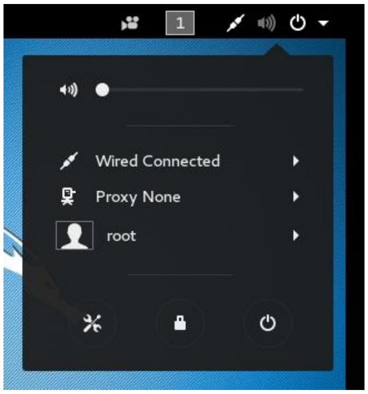
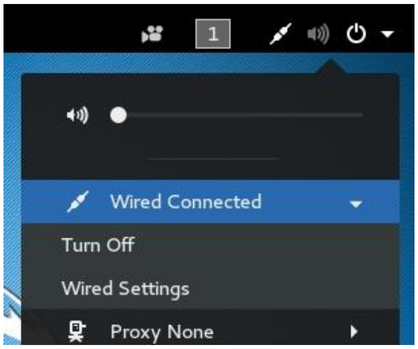
to verify the IP address has being set properly.

Set Internet Explorer to the same proxy settings, and attempt to access an external site to verify Internet connectivity.

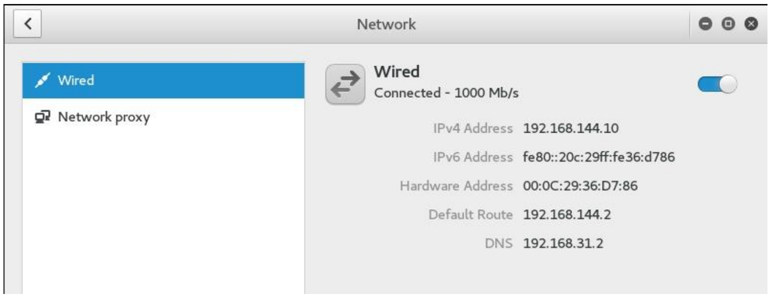
To set the IP address statically in the Kali Linux VM, login with the admin account

 Username: root	 Password: ••••

Then click on the Power icon, select `Wired Connected` -> `Wired Settings`:

From the `Main Network` dialog box, select the `Wired` dialog box,



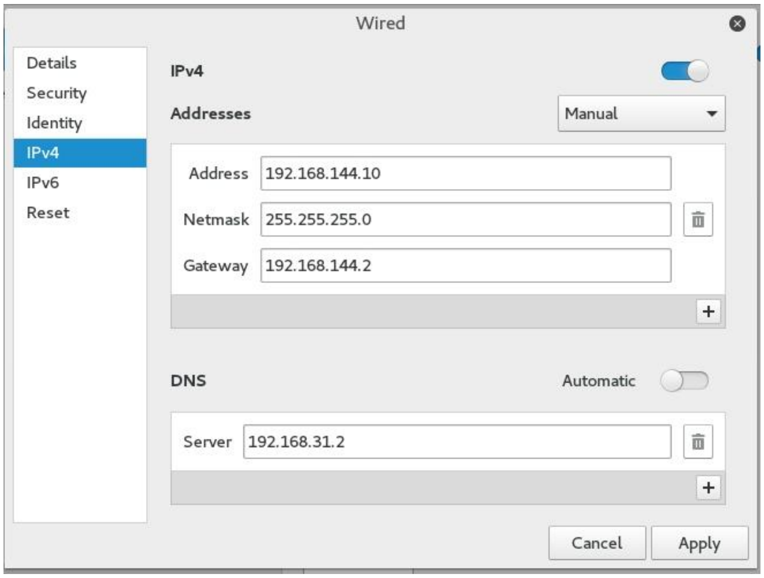
Wired
Connected - 1000 Mb/s

IPv4 Address	192.168.144.10
IPv6 Address	fe80::20c:29ff:fe36:d786
Hardware Address	00:0C:29:36:D7:86
Default Route	192.168.144.2
DNS	192.168.31.2

Then select `IPv4` and enter the following values:

`Kali Linux VM`

Address:	**129.168.144.10**
Netmask:	**255.255.255.0**
Gateway:	**192.168.144.2**
DNS Server:	**192.168.31.2**



IPv4 ●

Addresses: Manual

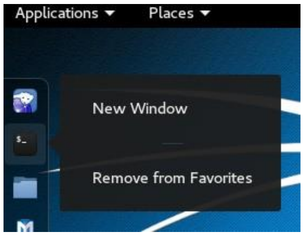
Address: 192.168.144.10
Netmask: 255.255.255.0
Gateway: 192.168.144.2

DNS — Automatic ○

Server: 192.168.31.2

When you are done, click Apply and close all open dialog boxes

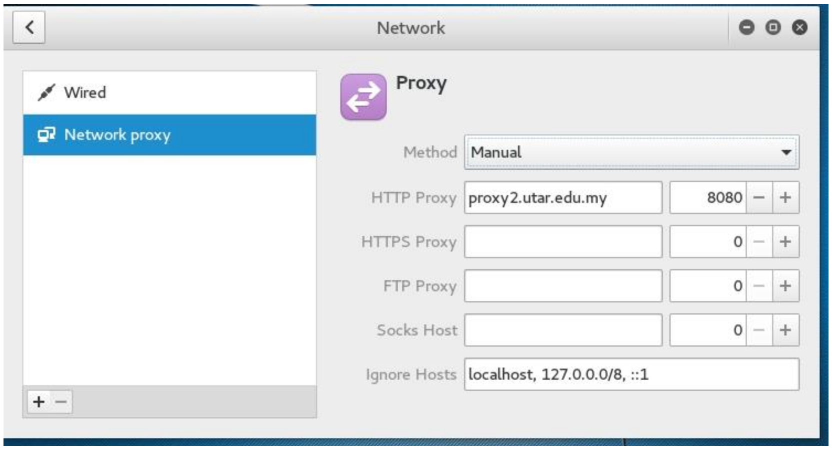
Open a Linux shell terminal, by right clicking on the Shell icon in the left menu bar and selecting New Window.



Type the following command to verify that the IP address has being set properly.

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet   192.168.144.10     netmask   255.255.255.0     broadcast
192.168.144.255
        inet6   fe80::20c:29ff:fe71:29e5     prefixlen   64      scopeid
0x20<link>
        ether 00:0c:29:71:29:e5  txqueuelen 1000  (Ethernet)
        RX packets 5141  bytes 7238982 (6.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 416  bytes 32067 (31.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 20  bytes 1200 (1.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20  bytes 1200 (1.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Then from the `Main Network` dialog box, select the `Network Proxy` dialog box to set the same settings for the university proxy server as in the case of the Windows VMs.



Method: Manual

HTTP Proxy: proxy2.utar.edu.my 8080

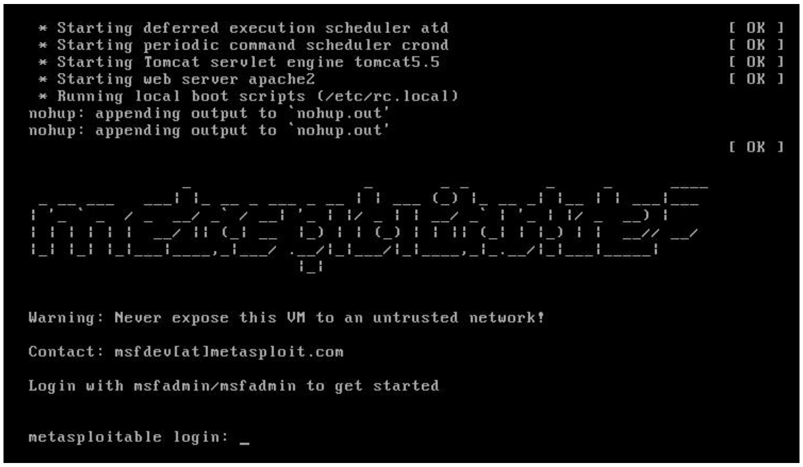
HTTPS Proxy: 0

FTP Proxy: 0

Socks Host: 0

Ignore Hosts: localhost, 127.0.0.0/8, ::1

To set the static IP address for the Metasploitable VM, login to it with the admin account. Metasploitable's login screen is a simple console display and you will interact with this VM through a single terminal shell, rather than a GUI-based desktop such as Kali.



```
 * Starting deferred execution scheduler atd                          [ OK ]
 * Starting periodic command scheduler crond                          [ OK ]
 * Starting Tomcat servlet engine tomcat5.5                           [ OK ]
 * Starting web server apache2                                        [ OK ]
 * Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out'
                                                                      [ OK ]

                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: _
```

We can use the vi command line editor to edit the file that is used to configure networking on the VM. Type at the prompt:

```
msfadmin@metasploitable:~$ sudo vi/etc/network/interfaces
```

Ensure that the contents of this file is as follows below. You can use the guide at this URL for appropriate vi commands: https://www.cs.colostate.edu/helpdocs/vi.html

```
auto eth0
iface eth0 inet static
address 192.168.144.40
netmask 255.255.255.0
network 192.168.144
broadcast 192.168.144.255
gateway 192.168.144.2
dns-nameservers 192.168.31.2 192.168.31.9
```

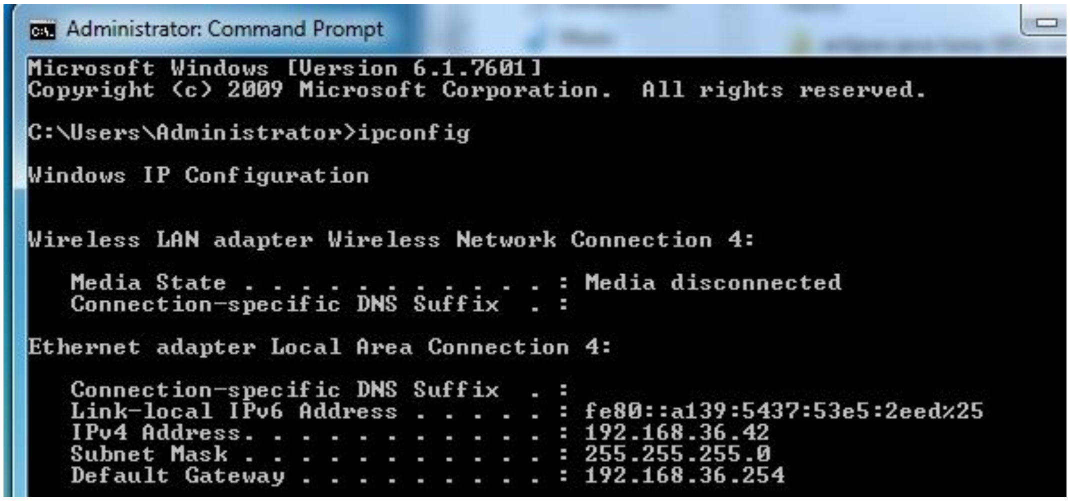
When you are done making the appropriate changes, exit and save the file and initialize with the new networking configurations:

```
msfadmin@metasploitable:~$ sudo /etc/init.d/networking restart
```

Then type the following command to verify that the IP address has being set properly:

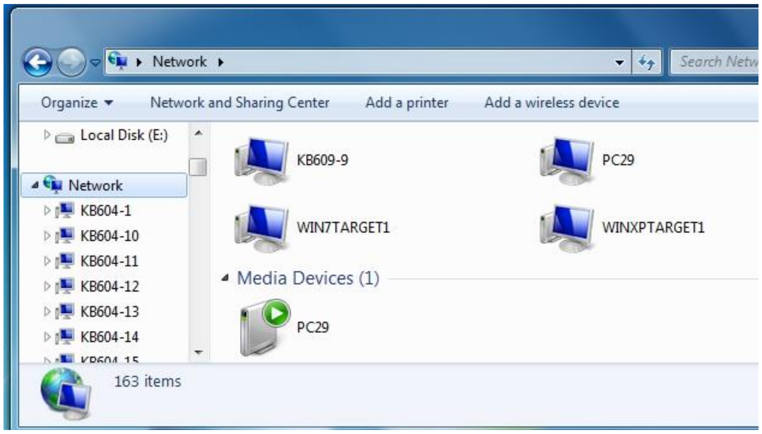
```
msfadmin@metasploitable:~$ ifconfig
eth0  Link encap:Ethernet HWaddr:00:0c:29:fa:dd2a
      inet      addr:192.168.144.40      Bcast:192.168.144.255      Mask:
255.255.255.0
………. .
………. .
```

Notice that the network address for the NAT is also different from the network address for the actual LAN that the physical machine running the guest OS is located in. You can open a command prompt in the host OS and type `ipconfig` to verify the network address for the actual LAN:



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP Configuration


Wireless LAN adapter Wireless Network Connection 4:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Ethernet adapter Local Area Connection 4:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::a139:5437:53e5:2eed%25
   IPv4 Address. . . . . . . . . . . : 192.168.36.42
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.36.254
```

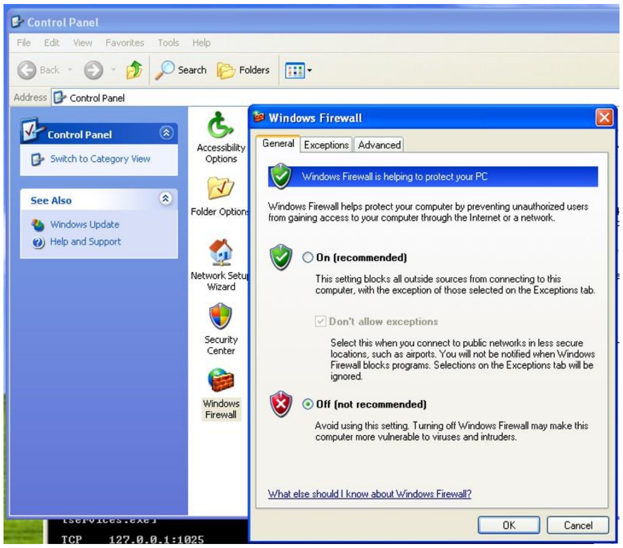
In the Network area of File Explorer, you should be able to view the machine names of both the Windows VMs in addition to all the other machines in the LAN. You will not however be able to view or access other machines connected to the physical network of the host machine from either of these 2 VMs.



4.5 Checking for connectivity between VMs

Before we can run any of the hacking exercises, we need to ensure that there is connectivity between all the VMs in the NAT virtual network. The simplest way to do this is to use the ICMP protocol to perform a ping between the active VMs running in NAT mode. For a ping to work, the firewall on both the Windows VMs need to be turned off. By default, they should already be turned off.

To work with the firewall in the Windows XP VM, go to `Control Panel -> Windows Firewall`. Select the appropriate radio button to turn on / off the firewall.



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

○ On (recommended)

This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.

☑ Don't allow exceptions

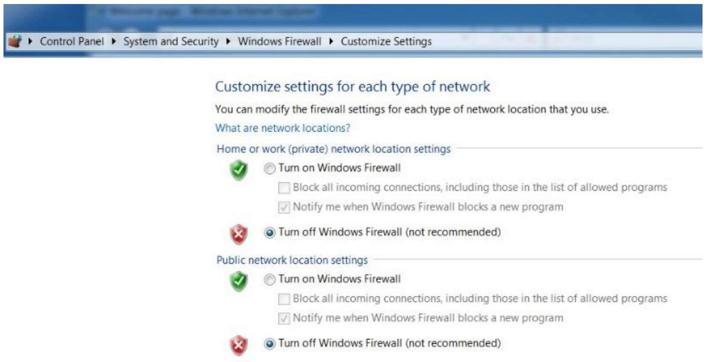
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.

● Off (not recommended)

Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.

What else should I know about Windows Firewall?

To work with the firewall in the Windows 7 VM, go to `Control Panel -> System and Security -> Windows Firewall` and select the appropriate radio button to turn on / off the firewall.



Control Panel ▸ System and Security ▸ Windows Firewall ▸ Customize Settings

Customize settings for each type of network

You can modify the firewall settings for each type of network location that you use.

What are network locations?

Home or work (private) network location settings

○ Turn on Windows Firewall

□ Block all incoming connections, including those in the list of allowed programs

☑ Notify me when Windows Firewall blocks a new program

● Turn off Windows Firewall (not recommended)

Public network location settings

○ Turn on Windows Firewall

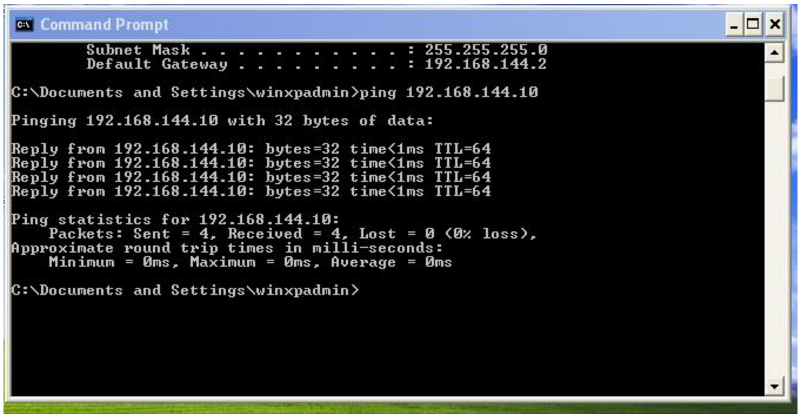
□ Block all incoming connections, including those in the list of allowed programs

☑ Notify me when Windows Firewall blocks a new program

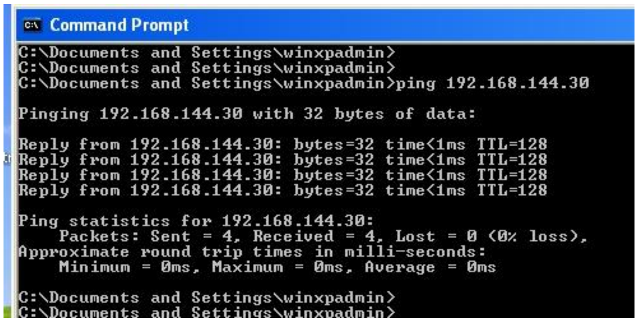
● Turn off Windows Firewall (not recommended)

Open a command prompt in both the Windows VMs and a shell terminal in the Kali Linux VM. Use a ping command to ping the two other VMs from each of these 3 VMs.

From within the Windows XP VM, we can attempt to ping the Windows 7 VM and Kali Linux VM

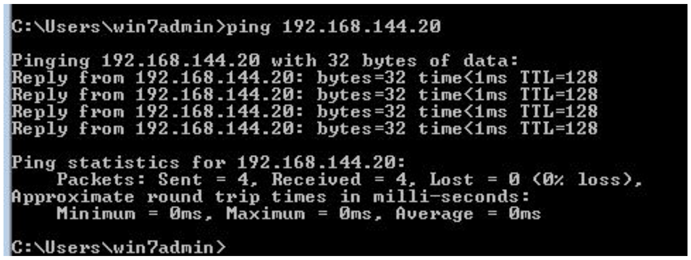


```
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.144.2

C:\Documents and Settings\winxpadmin>ping 192.168.144.10

Pinging 192.168.144.10 with 32 bytes of data:

Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.144.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\winxpadmin>
```

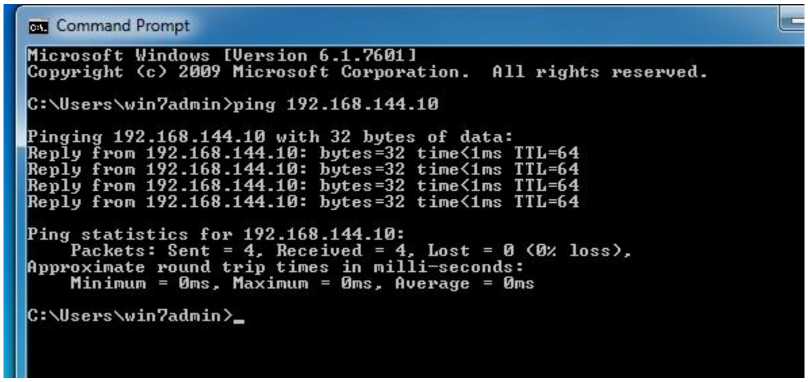


```
C:\Documents and Settings\winxpadmin>
C:\Documents and Settings\winxpadmin>
C:\Documents and Settings\winxpadmin>ping 192.168.144.30

Pinging 192.168.144.30 with 32 bytes of data:

Reply from 192.168.144.30: bytes=32 time<1ms TTL=128
Reply from 192.168.144.30: bytes=32 time<1ms TTL=128
Reply from 192.168.144.30: bytes=32 time<1ms TTL=128
Reply from 192.168.144.30: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.144.30:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\winxpadmin>
C:\Documents and Settings\winxpadmin>
```

From within the Windows 7 VM, we can attempt to ping the Windows XP VM and Kali Linux VM

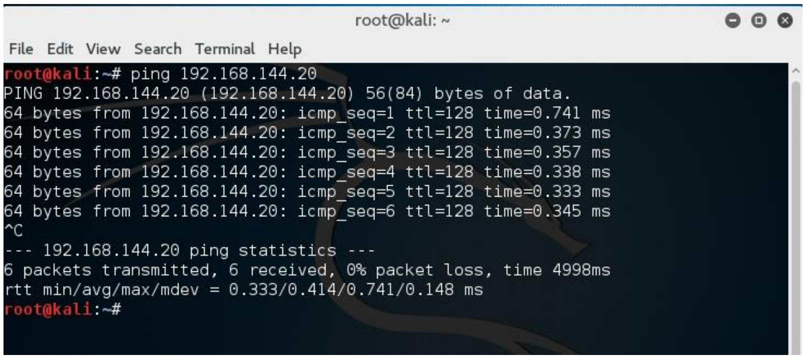


```
C:\Users\win7admin>ping 192.168.144.20

Pinging 192.168.144.20 with 32 bytes of data:
Reply from 192.168.144.20: bytes=32 time<1ms TTL=128
Reply from 192.168.144.20: bytes=32 time<1ms TTL=128
Reply from 192.168.144.20: bytes=32 time<1ms TTL=128
Reply from 192.168.144.20: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.144.20:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\win7admin>
```

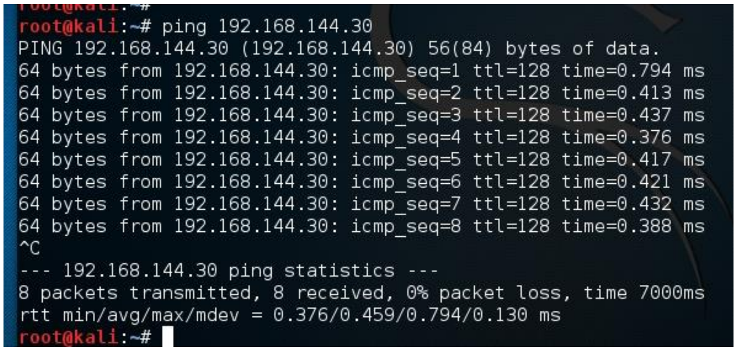


```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\win7admin>ping 192.168.144.10

Pinging 192.168.144.10 with 32 bytes of data:
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64
Reply from 192.168.144.10: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.144.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\win7admin>_
```

From within the Kali Linux VM, can attempt to ping both the Windows VMs. The ping command in Linux continues to run indefinitely, so you can terminate it by typing Ctrl-C once you have received a few acknowledgments of success.



```
root@kali:~# ping 192.168.144.20
PING 192.168.144.20 (192.168.144.20) 56(84) bytes of data.
64 bytes from 192.168.144.20: icmp_seq=1 ttl=128 time=0.741 ms
64 bytes from 192.168.144.20: icmp_seq=2 ttl=128 time=0.373 ms
64 bytes from 192.168.144.20: icmp_seq=3 ttl=128 time=0.357 ms
64 bytes from 192.168.144.20: icmp_seq=4 ttl=128 time=0.338 ms
64 bytes from 192.168.144.20: icmp_seq=5 ttl=128 time=0.333 ms
64 bytes from 192.168.144.20: icmp_seq=6 ttl=128 time=0.345 ms
^C
--- 192.168.144.20 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 4998ms
rtt min/avg/max/mdev = 0.333/0.414/0.741/0.148 ms
root@kali:~#
```



```
root@kali:~#
root@kali:~# ping 192.168.144.30
PING 192.168.144.30 (192.168.144.30) 56(84) bytes of data.
64 bytes from 192.168.144.30: icmp_seq=1 ttl=128 time=0.794 ms
64 bytes from 192.168.144.30: icmp_seq=2 ttl=128 time=0.413 ms
64 bytes from 192.168.144.30: icmp_seq=3 ttl=128 time=0.437 ms
64 bytes from 192.168.144.30: icmp_seq=4 ttl=128 time=0.376 ms
64 bytes from 192.168.144.30: icmp_seq=5 ttl=128 time=0.417 ms
64 bytes from 192.168.144.30: icmp_seq=6 ttl=128 time=0.421 ms
64 bytes from 192.168.144.30: icmp_seq=7 ttl=128 time=0.432 ms
64 bytes from 192.168.144.30: icmp_seq=8 ttl=128 time=0.388 ms
^C
--- 192.168.144.30 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7000ms
rtt min/avg/max/mdev = 0.376/0.459/0.794/0.130 ms
root@kali:~# 
```

You can repeat the same procedure to establish connectivity between the Metasploitable VM and the 3 other VMs.

Always ensure you check for connectivity between all the VMs involved in a lab BEFORE starting the exercises. Inability to send pings between two VMs running in NAT mode is usually due to the static IP addresses of one of the VMs being set incorrectly.