Lab E5: Network Sniffing

TTM4175

September 28, 2015

In this lab you will be looking at ways to sniff traffic that goes over a network. You will be setting up a virtual local network inside VirtualBox and using a packet capture program called Wireshark from within Kali Linux to analyze the network traffic. By our set up Kali will be able to see all traffic that passes over the network – even packets that are not addressed to it – this will simulate the state of affairs on a wireless network where an attacker can listen in on everything. However, we will also show how you can obtain network traffic on the Local Area Network (LAN) if you are not using wireless.

1 Setting up the network

Lab E1 showed you how to set up a virtual local area network (LAN) within VirtualBox. This lab will build on that and expand the network by one machine, giving a network as shown in Figure 1. Before you continue, ensure that your Host-only network is set up as specified in Lab E1, and that both your Kali and Windows machines connect to it.

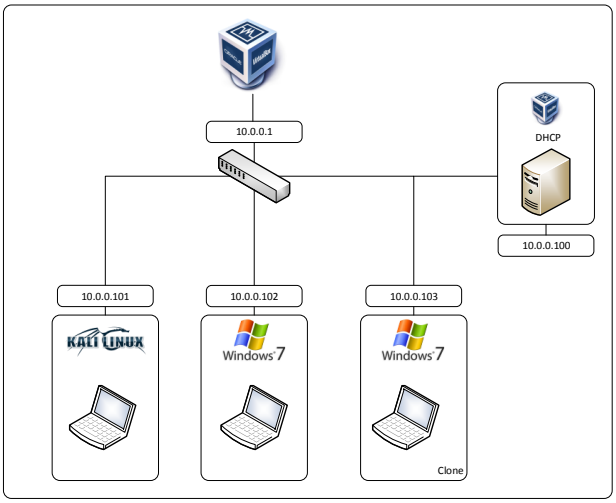


Figure 1: Network layout for Lab E5.

Now, create a clone of your Windows 7 machine. It is important that you enable the option "Reinitialize the MAC address of all network cards" (Figure 2) so that no computers on the network ends up with the same MAC address. Let the clone be of type "Linked clone" which is faster to create and also saves a bit of hard disk space on your host computer.

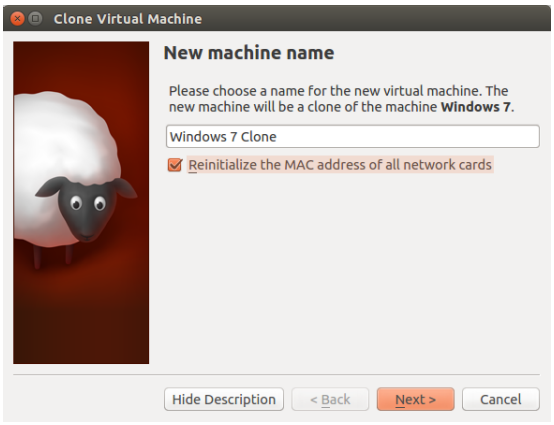


Figure 2: Give a new MAC addresses to the Windows 7 clone.

When the clone is created, ensure that its network mode is set to "Host-only adapter" and that selected adapter matches that of the other machines.

Note: If you in a previous lab changed the amount of RAM and number of processors assigned to your Kali Linux it might be a good idea to change this back now. That is, your Kali Linux machine should have no more than 4GB of RAM assigned, and not more than 1-2 processors ("Settings → System → Motherboard/Processor"). Similarly, the Windows machines only needs one 1 CPU and can live with as little as 1024 MB of RAM.

Start up all three machines, and log in as user `ttm4175` on the Windows machines (if you did not manage to obtain this password in Lab E3, it's "`1234`"). Verify that all the machines are correctly assigned IP addresses in the `10.0.0.101`–`10.0.0.200` range. For the rest of this lab description we assume that the machines have been given the IP addresses as shown in Figure 1, that is:

- Kali Linux: `10.0.0.101`;
- Windows 7: `10.0.0.102`;
- Windows 7 Clone: `10.0.0.103`;
- DHCP server: `10.0.0.100`;
- Default Gateway: `10.0.0.1`.

Note: Your machines might get assigned different IP addresses then those shown above. Make sure to account for this when reading the instructions below!

2 Sniffing network traffic with Wireshark

Eavesdropping generically refers to intentionally listening in on a private conversation. In the context of computer networks eavesdropping is often called *sniffing*, and involves capturing and decoding network packets on the network which are not addressed to your machine.

The usefulness of sniffing is immediately obvious when the data is being passed in plain text, because any "cleartext" data sniffed is data that can be immediately read. Usernames and passwords are often easy to extract with a little network protocol knowledge. This exactly what you will be doing now: you will be listing in on a session using one of the oldest communication protocols on the Internet, namely *Telnet*. It was developed as early as 1968 and is still used on the Internet today! Unfortunately, the 60s was a happier time and everything was sent in the clear back then. This makes Telnet an easy target to eavesdrop on. While it is no longer used that often for sensitive Internet traffic, it is still used in many internal networks (like companies and banks) because of legacy systems that cannot handle encryption. Also, many wireless home routers allows you to connect to their

2.1 Simulating a wireless network: promiscuous mode

Before we begin, we need to make sure that the Kali machine can see all traffic that passes over the network. In cabled networks that use an *Ethernet switch* to connect the computers together (which VirtualBox simulates by default), only the correct recipient of a packet will receive it. This means that the Kali machine will only see packets that are addressed to itself and not the packets going between the Windows machines. However, within VirtualBox we can modify this slightly by changing Kali's network attachment into so-called *promiscuous mode*. This will allow Kali too see all packets going over the LAN, even those not addressed to it.

Go to the network settings of your Kali machine and click "Advanced" in order to see all the options. Change the value of "Promiscuous Mode" from "Deny" to "Allow All" (Figure 3).

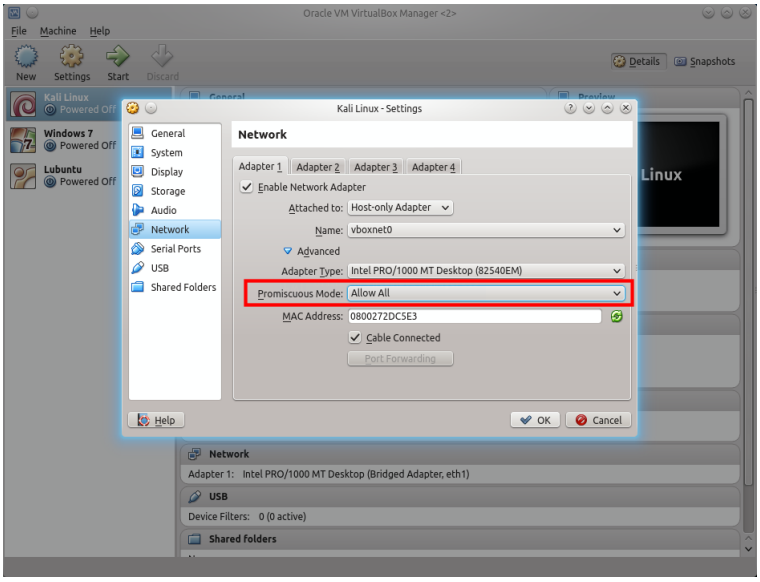


Figure 3: Enabling promiscuous network mode on the Kali machine.

Lest you believe that this scenario is too advantageous for the attacker, remember that it corresponds exactly to the way a wireless network works, where every packet can be seen by everyone.

2.2 Enabling Telnet on Windows 7

Telnet allows a user to connect remotely to other machines over a network. Telnet comes pre-installed on Windows 7, but is disabled per default. To enable it, do the following while being logged into the Windows 7 machines:

On both machines (original and clone). Open up the Start Menu and go to "Control Panel → Programs → Turn Windows feature on or off" and scroll down to until you find "Telnet Client" (Figure 4). Enable it and click "OK".

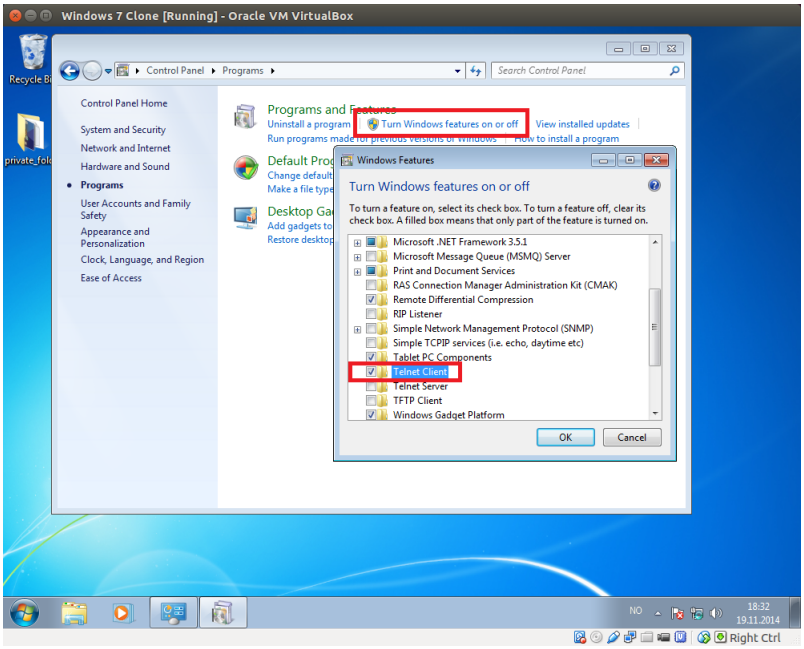


Figure 4: Enabling the Telnet Client on Windows 7.

Only on the Windows 7 Clone. In addition to enabling the Telnet Client, which allows you to connect *to other* machines, we will also turn on Telnet Server which allows other machines to connect *to us*. It is enabled at the same place as the Telnet Client. Additionally, after having enabled the Telnet Server feature, we also have to turn it on as a *service*. A Windows service is program that runs in the background of your computer and can be scheduled to be run at certain time intervals or at start up. From the Start Menu type '`services`' and hit Enter. Scroll down until you find "Telnet", double-click it and change "Startup type" to "Automatic" in the scroll-down window; then finish with "Apply". Start the Telnet server by clicking "Start" (Figure 5).

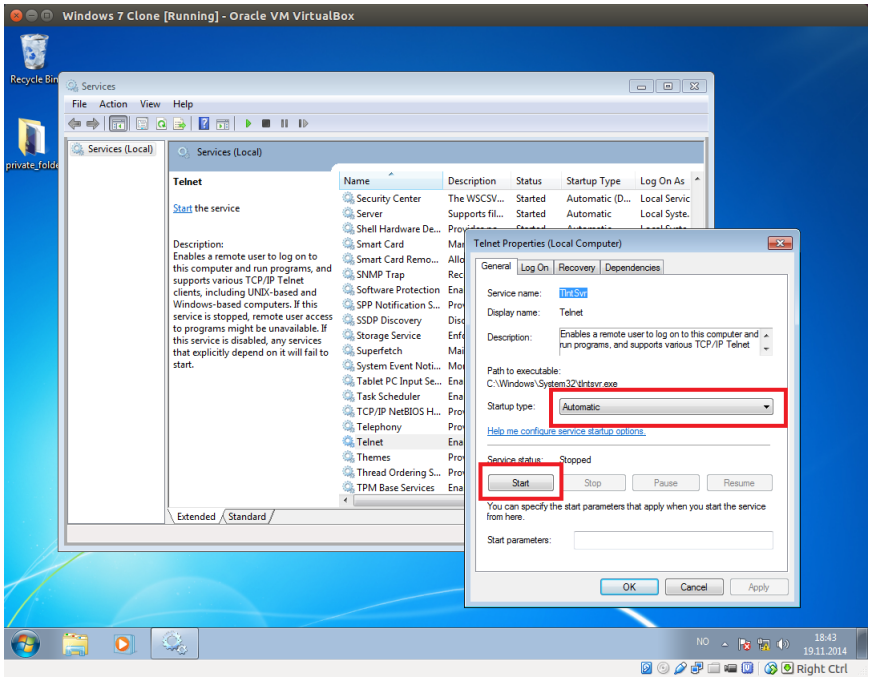


Figure 5: Starting the Telnet Server service on the Windows 7 Clone.

The Windows 7 Clone now listens for incoming Telnet connections on port 23. However, in order for a remote connection to have access to log in as a certain user, we need to add an account to the *TelnetClients group*. On the Windows 7 Clone, start up the command line with administrator rights ("Start Menu → type `cmd` → right-click the icon and select "Run as administrator"") and execute the following:

```
C:\Windows\system32> net localgroup TelnetClients /add ttm4175
C:\Windows\system32> tlntadmn config sec -ntlm +passwd   % require a password when logging in
C:\Windows\system32> tlntadmn config maxconn=50          % allow up to 50 simultaneous connections
```

If successful then people can now access the `ttm4175` account remotely with Telnet using its password.

Note: Telnet requires there to be a password when logging in remotely, so if you wiped the password of the `ttm4175` account in Lab E2 or E3, make sure to set a password now.

2.3 Scanning open ports with Nmap

Before starting the sniffing we will verify that the Windows 7 Clone is actually listening for incoming Telnet connections. While we could of course test this directly with the Windows machine, we will instead use the opportunity to introduce a *port scanner* called `Nmap`. A port scanner is program that looks for open (listening) ports on a machine that resides on a network. This is typically among the first steps – often called the *reconnaissance phase* – an attacker will carry out before attempting to attack a target machine.

By enumerating the open ports of a machine the attacker can learn which services are running on it, and hopefully find someone with a known vulnerability. Conversely, if for example the attacker finds that no Telnet service is running on a machine, there is no point in trying a Telnet exploit against it.

Nmap is a very powerful program, but we will only use it to establish that Telnet is actually running on the Windows 7 Clone. From the terminal in Kali run the following command (you can use the `-v` flag if you want even more output):

```
% Remember that while our Clone is on 10.0.0.103 it could be different for you!
# nmap -sV --top-ports 20 10.0.0.103 --system-dns

Starting Nmap 6.47 ( http://nmap.org ) at 2014-09-04 12:17 CEST
Nmap scan report for 10.0.0.103
Host is up (0.00063s latency).
PORT     STATE    SERVICE       VERSION
21/tcp   filtered ftp
22/tcp   filtered ssh
23/tcp   open     telnet        Microsoft Windows 2000 telnetd
25/tcp   filtered smtp
53/tcp   filtered domain
80/tcp   filtered http
110/tcp  filtered pop3
      ...
```

This command scans a target for the 20 most commonly used ports and checks whether they are listening or not. As can be seen from the listing above the Windows 7 Clone is indeed listening on port 23, which is the Telnet service. If you want to try out some other network scans with Nmap type `nmap -h | less` to get a list of available options.

2.4 Starting the network sniffing

Now we are ready to start capturing network traffic. We will use a network analyzer called *Wireshark* which is available for both Windows, OSX and Linux, and comes pre-installed with Kali Linux. Wireshark is an extremely powerful program that lets you observe network traffic in real-time with a graphical user interface. Start up Wireshark in Kali (either from the terminal: just type '`wireshark &`'; or from the applications menu: "Applications → Internet → Wireshark") and dismiss the warnings about running as root[1]. After Wireshark has started click on "Capture Options". From this menu we can configure several options, like which network interface we want to capture traffic on (modern computers usually have multiple network cards, including Ethernet and Wi-Fi) and in what mode. Select the `eth0` interface, and ensure that promiscuous mode is enabled. Moreover, disable the "Resolve MAC addresses" option. You should have settings similar to those in Figure 6.

[1]Running Wireshark as root is not a best-practice, but is unproblematic in this lab. However, if you ever run Wireshark on your own computer, it would be wise to heed this warning.

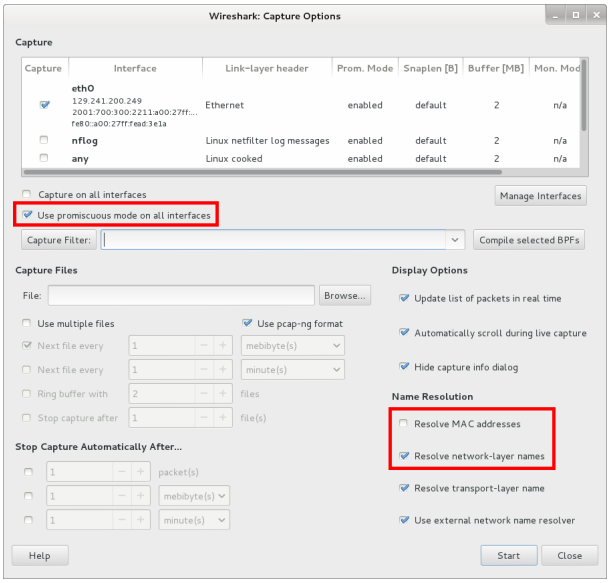


Figure 6: Configuring the capture options in Wireshark.

Start the capture, and soon you should see packets flowing in like in Figure 7. By selecting a specific packet you can get more protocol information about it.

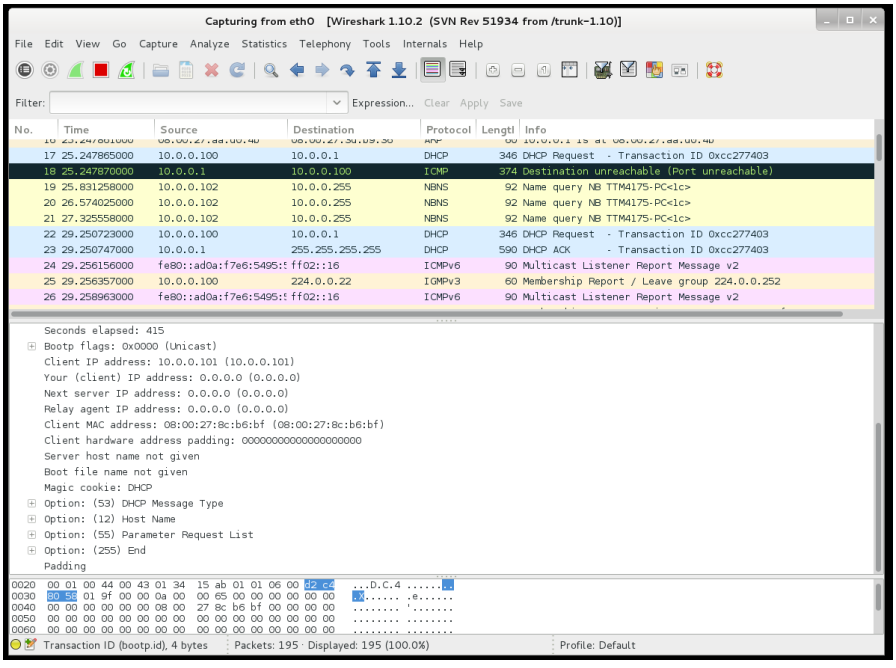


Figure 7: Wireshark in action.

If you do not receive any packets it might be because you haven't turned on promis-

cuous mode. Remember to do this both in Wireshark (as described above) and in the VirtualBox network settings of your Kali Linux machine (as described in Section 2.1).

Filters. You will soon realize that the amount of packets you receive is larger than what you can easily work with. Fortunately, in Wireshark you can apply *filters* that will remove some of the packets. There are two types of filters in Wireshark: *capture* filters and *display* filters. Capture filters affects the actual packets that Wireshark will capture and process. If a filter does not apply to a packet it will simply drop it. Display filters on the other hand does not influence which packets Wireshark captures or not, only which one it displays. We will only be using display filters.

In the "Filter" menu of Wireshark type in `icmp` and hit Enter. This will only display packets of the ICMP[2] type. On one of your Windows machines, open the command line and ping the default gateway on the network:

```
C:\Users\ttm4175> ping 10.0.0.1
```

If everything is set up correctly you should be able to see these messages within Wireshark on your Kali machine (Figure 8).

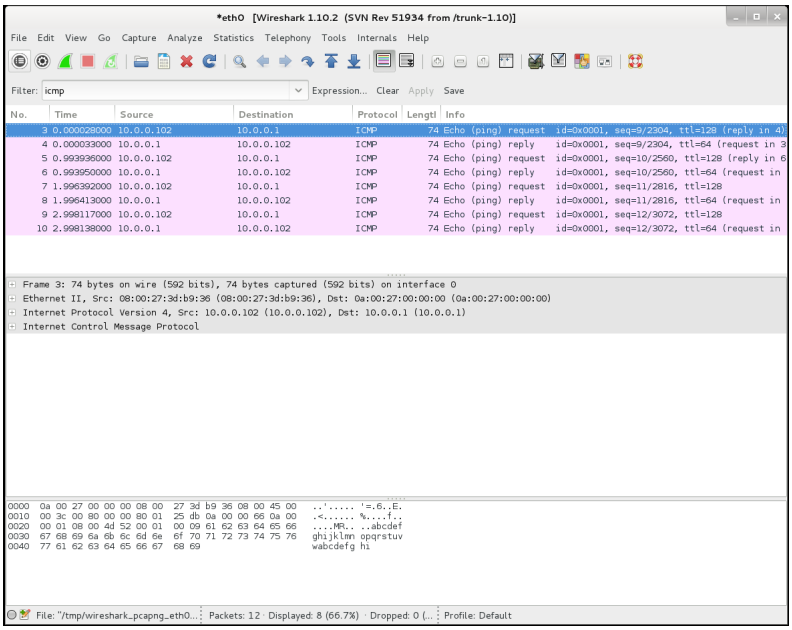


Figure 8: Capturing a ping from the Windows 7 machine (`10.0.0.102`) to the default gateway (`10.0.0.1`) with Wireshark running on your Kali Linux machine.

You are now effectively eavesdropping on the network. Leave Wireshark open for the rest of this lab, capturing everything on the network.

[2]Internet Control Message Protocol — one of the oldest network protocols on the Internet. Used to transfer error messages between network devices. Typically used by the `ping` utility to check whether a host is up or not.

2.5 Capturing a Telnet session

First change the filter from `icmp` to `telnet` in Wireshark so that you only display Telnet packets. Then on your Windows 7 machine (*not* the Clone!) open up the command line and type `telnet`:

```
C:\Users\ttm4175> telnet

Welcome to Microsoft Telnet Client

Escape Character is 'CTRL+¨'

Microsoft Telnet>
```

Type '`open 10.0.0.103`' to initiate a connection to the Windows 7 Clone. If prompted for a confirmation type '`y`' to allow the connection:

```
Microsoft Telnet> open 10.0.0.103
Connecting To 10.0.0.103...
You are about to send your password information to a remote computer in Internet
 zone. This might not be safe. Do you want to send anyway(y/n): y
```

When prompted for login credentials, type in `ttm4175` and the password you obtained from Lab E3:

```
Telnet server could not log you in using NTLM authentication.
Your password may have expired.
Login using username and password

Welcome to Microsoft Telnet Service

login: ttm4175
password:


Microsoft Telnet Server.
*===============================================================
C:\Users\ttm4175>
```

You are now logged in to the Windows 7 Clone via Telnet. Type `dir` to see the files on the machine:

```
C:\Users\ttm4175>dir
 Volume in drive C has no label.
 Volume Serial Number is 0C19-58D2

 Directory of C:\Users\ttm4175

27.08.2014  15:56    <DIR>          .
27.08.2014  15:56    <DIR>          ..
06.08.2014  16:11    <DIR>          Contacts
01.09.2014  14:10    <DIR>          Desktop
06.08.2014  16:11    <DIR>          Documents
         ...
06.08.2014  16:11    <DIR>          Videos
               0 File(s)              0 bytes
              13 Dir(s)  1 059 926 880 bytes free

C:\Users\ttm4175>
```

These are the files residing on the *Clone* —not the original Windows machine. Head over to your Kali machine and look at what you have captured with Wireshark. Hopefully you should have captured the entire session (Figure 9). Can you find the username and password that was used to log in (see the hint below)?

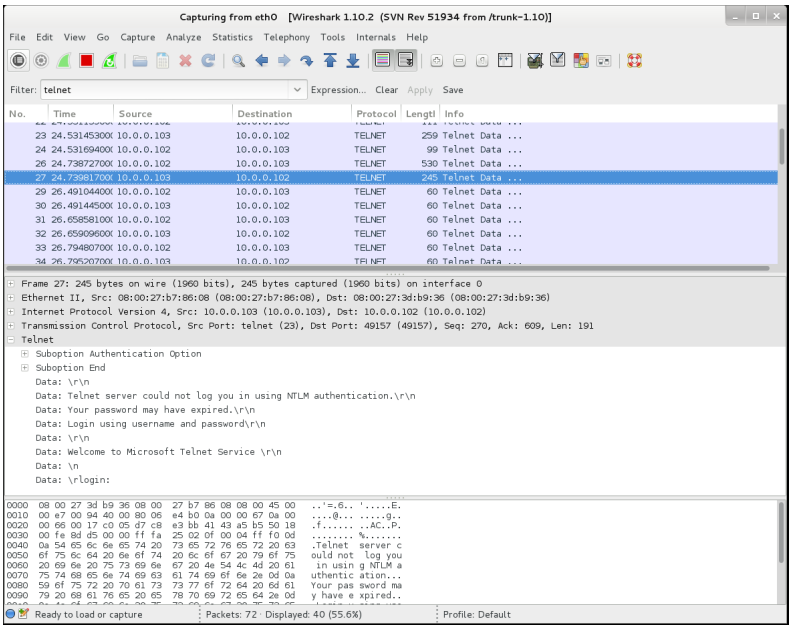


Figure 9: Capturing a Telnet session between the two Windows machines in Wireshark.

Hint: Telnet is a *character oriented* protocol, which means that it will send your characters over the connection as soon as you type them in. This means that the username and password that you used to log in were probably spread over multiple Telnet packets. This is a bit different from other remote terminal protocols where the commands are *buffered* until you hit the Enter key, in which case it sends everything in a single packet.

To illustrate that *everything* goes in the clear over Telnet, not just usernames and passwords, read a file over Telnet an sniff its contents using Wireshark. On the Clone machine, first create a simple text document `test.txt` on the Desktop, and a few lines of text to it. Now, in the Telnet window of *original* Windows machine, change to the `Desktop` folder and print the contents of the file `text.txt` using the command `type` (`type` is similar to the `cat` command in Linux).

```
C:\Users\ttm4175> cd Desktop
C:\Users\ttm4175> type test.txt
```

What do you observe in Wireshark? After you are finished leave the remote Telnet session by typing `exit` and hitting Enter.

2.6 Man-in-the-middle on a *wired* LAN – ARP spoofing

So far we have simulated that you have been running on a wireless LAN, which makes eavesdropping trivial. Now we will see how we can still sniff network traffic, even if we are connected to a *wired* LAN.

Start by turning off promiscuous on your Kali Linux machine: "Settings → Network → Promiscuous Mode: Deny". Next stop the running Wireshark capture by clicking on the red square then save the captured packets to `part2_5.pcapng` by clicking "File → Save as...". Now start a new live capture by clicking on the green shark fin next to the stop button labeled "Start a new live capture".

On your original Windows machine log into the Clone using Telnet again. What do you see in Wireshark now? Kali Linux no longer receive any packets from the Windows machines since you have turned off promiscuous mode. Basically the Kali machine has been isolated on the network. Leave the Telnet session again by typing `exit`.

If you open up a new command line window on the Windows machine and type in '`arp -a`' you will see the current ARP[3] table that this machine has (remember that the original Windows machine has IP address `10.0.0.2` in this lab description):

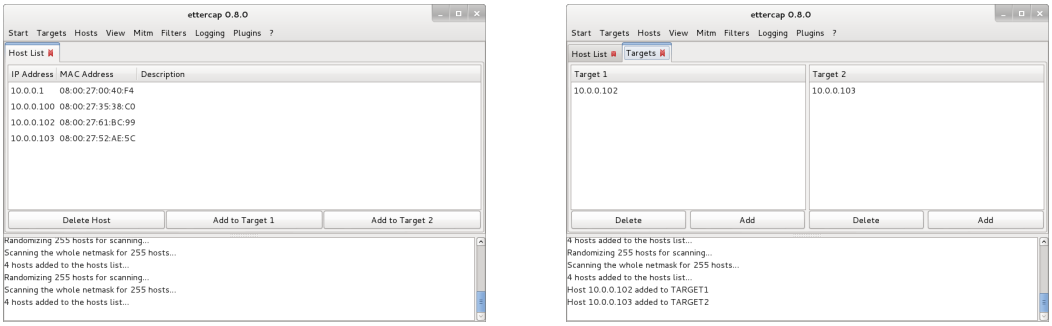
```
C:\Windows\system32> arp -a

Interface: 10.0.0.102 --- 0xb
  Internet Address      Physical Address      Type
  10.0.0.100            08-00-27-35-38-c0     dynamic   % IP/MAC address of DHCP server
  10.0.0.101            08-00-27-ad-3e-1a     dynamic   % IP/MAC address of Kali Linux machine
  10.0.0.103            08-00-27-52-ae-5c     dynamic   % IP/MAC address of Windows Clone
                  ...
```

This table shows the MAC address of the machine have IP address `X`. Note that all the IP addresses are mapped to the correct MAC addresses (verify this by checking the MAC addresses of your virtual machines! On a Windows machine you can find this information by typing '`ipconfig /all`' on the command line, while on a Linux machine you would type '`ifconfig`'). Now we will launch an *ARP-spoofing* attack against the Windows machines which will trick them into sending their traffic to the Kali Linux machine instead. We will use a potent program called *Ettercap* that also has an easy-to-use graphical interface. Start it in Kali Linux by running:

```
# ettercap -G &
```

Once started, begin by selecting "Sniff → Unified sniffing" and accept the choice of `eth0` as the network interface to sniff on. First we perform a scan of the LAN to find all the hosts on it. Go to "Hosts" and select "Scan for hosts". If you now go to "Hosts → Hosts list", all the entities on the network should be listed (Figure 10a).

[3]Address Resolution Protocol — the protocol used to discover the MAC address of a computer having a given IP address on a LAN.



(a) Hosts on LAN. **(b)** Selected targets.

Figure 10: Running Ettercap.

Select `10.0.0.102` and click on "Add to Target 1", similarly add `10.0.0.103` as Target 2. If you click on "Targets → Current Targets" you will see that they have been added as targets for our ARP-spoofing attack (Figure 10b).

Now it is time to start the ARP-attack in order to become a man-in-the-middle. Click on "Mitm → Arp poisoning[4]... →", then enable "Sniff remote connections" and click "OK". The attack has been launched. If you now run `arp -a` on one of the Windows machines, what do you observe? For example, on the original Windows machine:

```
C:\Windows\system32> arp -a

Interface: 10.0.0.102 --- 0xb
  Internet Address      Physical Address      Type
  10.0.0.100            08-00-27-35-38-c0     dynamic   % DHCP server
  10.0.0.101            08-00-27-ad-3e-1a     dynamic   % Kali Linux machine
  10.0.0.103            08-00-27-ad-3e-1a     dynamic   % Windows Clone IP,
                                                           but Kali Linux MAC address!
```

Notice that the attack changed the ARP table of the Windows machines. What will happen now if, e.g., the original Windows machine wanted to send a packet to IP address `10.0.0.103`? (see also Question 2) It will get sent to the Kali Linux machine instead!

Unfortunately, instead of forwarding the packets it receives, our Kali Linux machine will simply hold on to the traffic and not send it forward. Thus, the victims would probably soon notice that something was wrong, since all their packets would be lost (try it out with Telnet!). To change this behavior we need to enable *packet forwarding*. This is controlled by the file `/proc/sys/net/ipv4/ip_forward`, which contains the single character '`0`'; meaning that it will *not* forward packets. If we change it to '`1`', then Kali Linux will begin to forward packets, hence:

```
echo 1 > /proc/sys/net/ipv4/ip_forward   % set the ip_forward flag to 1 (1 == TRUE)
```

Now Kali Linux will function as a proper man-in-the-middle. Once again, log in using Telnet on your Windows machine. If you now switch to Wireshark you will see that the Telnet traffic is successfully captured from the *wired* LAN!

[4]ARP-spoofing attacks are often also called *ARP-poisoning* attacks.

Note: Before continuing to the next section be sure to stop the man-in-the-middle attack and reset the ARP-tables of the Windows machines to their original values. First exit the Telnet session on the Windows machine. Then in Ettercap, click on "Mitm → Stop mitm attack(s)" and give it time to finish. When its finished, close Ettercap and verify that the ARP-tables on the Windows machines have been properly reset using `arp -a`. Finally, we disable the IP forwarding by running

```
echo 0 > /proc/sys/net/ipv4/ip_forward
```

Additionally, *turn promiscuous mode back on* in the VirtualBox network settings for your Kali Linux machine. This way you do not have to do the ARP-spoofing technique above in order to capture the traffic in the rest of the lab.

Note: Make sure that you actually *close* the Ettercap process/window.

2.7 Modern remote login: SSH

Hopefully, the previous sections have convinced you that if you want to securely log in to a remote computer you should *not* use Telnet! Instead, these days most people use the *Secure Shell* (SSH) protocol. SSH functions very much like Telnet, however, it employs cryptography in order to protect your communication.

2.7.1 Starting a SSH server on your Windows 7 Clone

On the Windows machines we have installed an SSH server program called *freeSSHd*. On your Windows 7 *Clone* click on "Start → freeSSHd". The program starts and adds a little icon to the bottom right tray bar (Figure 11).

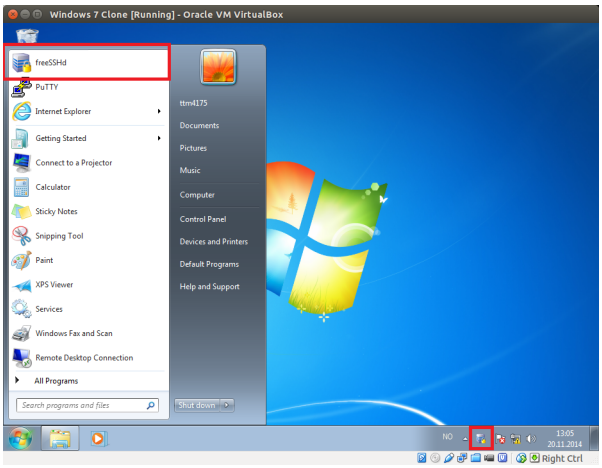


Figure 11: Starting freeSSHd.

We now have to configure the SSH login to be based on the users local computer password. Click on the freeSSHd icon down on the right to open the administration pane. Pick the "Users" tab and select the `ttm4175` user, then click on "Change" (Figure 12).

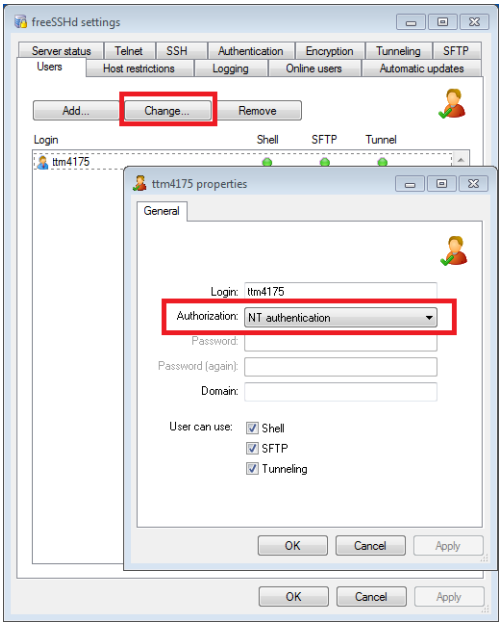


Figure 12: Configuring SSH to use the accounts local NT password for authentication.

Change the "Authorization" field from "Password stored as SHA1 hash" to "NT authentication". This will make SSH use the user's local NTLMv2 password hash, stored in the SAM database, to authenticating to the server (c.f. Lab E2).

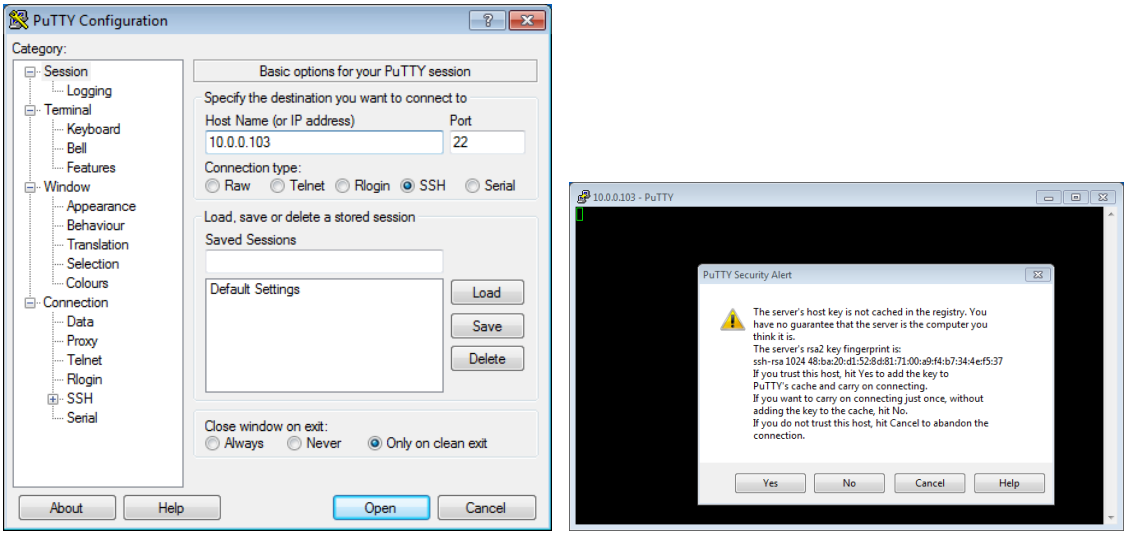
We can verify that the SSH server has started at the Clone through Kali Linux by running `nmap` again:

```
# nmap -sV --top-ports 20 10.0.0.103 --system-dns

Starting Nmap 6.47 ( http://nmap.org ) at 2014-09-04 12:17 CEST
Nmap scan report for 10.0.0.103
Host is up (0.00063s latency).
PORT     STATE    SERVICE       VERSION
21/tcp   filtered ftp
22/tcp   open     ssh           WeOnlyDo sshd 2.1.3 (protocol 2.0)
23/tcp   open     telnet        Microsoft Windows 2000 telnetd
25/tcp   filtered smtp
      ...
```

Notice that the status of port 22 (the default port for SSH) has changed from `filtered` to `open`, meaning that the Windows clone is listening for incoming connections on this port.

2.7.2 Logging in with SSH

With the SSH server started on the Clone, we will now use the *other* Windows machine to connect remotely to it using SSH. The SSH client program that we will use is called *PuTTY*. On your *non-clone* Windows 7 machine click on "Start" and select "PuTTY". In the input field labeled "Host Name" type in the IP address of *your* Windows 7 Clone and click on "Open" (figure 13a). The first time you connect you will be shown an alert message similiar to the one in Figure 13b. Just dismiss it by clicking "Yes" and you will be taken to a login prompt where you can provide the credentials of the `ttm4175` user.



(a) PuTTY settings. **(b)** SSH connection alert.

Figure 13: Remote SSH connection using PuTTY.

After you have successfully logged in you will be given a command line interface identical to the one you got with Telnet.

If you now turn to Wireshark on your Kali Linux machine and change the display filter to `ssh` you should see something similar to Figure 14 (this requires that you left Wireshark capture running, if not, restart the capture).

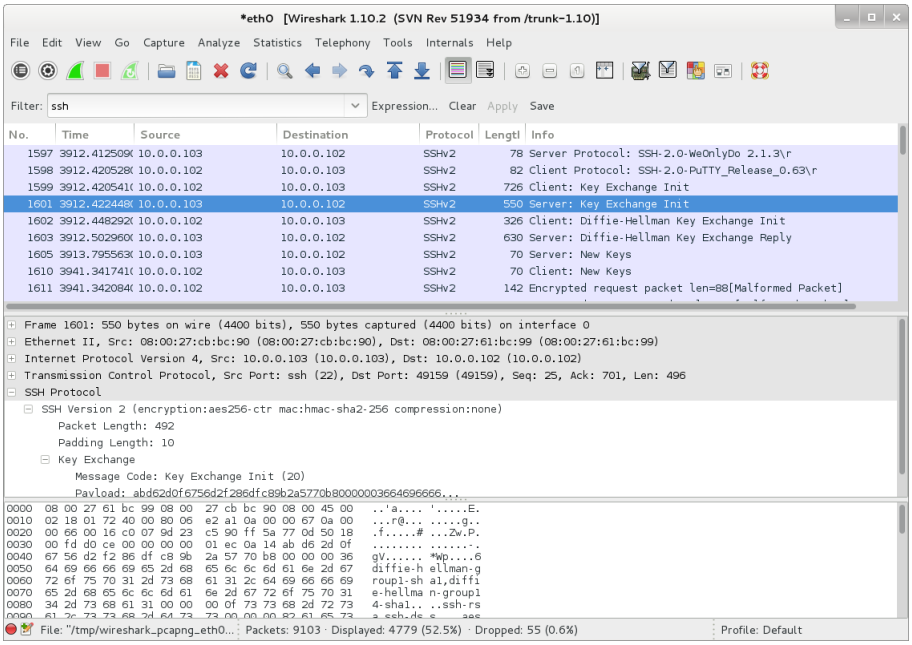


Figure 14: Capturing SSH traffic in Wireshark.

Are you able to find the user credentials that you used to login somewhere in this listing? (**Hint:** No.) What do you see if you repeat the experiment of printing the contents of the file `test.txt` as you did in Section 2.5?

3 Web sniffing and session hijacking

In this section you will see how easy it is to capture user credentials from poorly protected web sites. You will also be doing *session hijacking* which means that you will be taking over another person's ongoing session on a web page. All by just sniffing their traffic in Wireshark.

3.1 Switching to Bridged-mode in VirtualBox

Recall that the Host-only networking mode that you have been using so far (Figure 1) is completely separated from the outside world. In order to be able to access a web site on the Internet we need to change our network settings from Host-only mode to Bridged mode. The Bridged mode will allow your virtual machines to connect to the Internet, while still giving you the option of forwarding the packets to your Kali machine (so that you can sniff it).

In the VirtualBox settings of your Kali machine, go to "Network" and change the network mode into "Bridged Adapter" (Figure 15). For the "Name" option select the interface on which your *host* computer is connected to the Internet. The lab computers have two network interface cards: `eth0` and `eth1`. In Figure 15 the lab computer is

connected to the Internet through the `eth1` interface. Additionally, ensure that "Promiscuous Mode" is set to "Allow VM" so that Kali will receive a copy of all the traffic that pass through the adapter from the virtual machines.

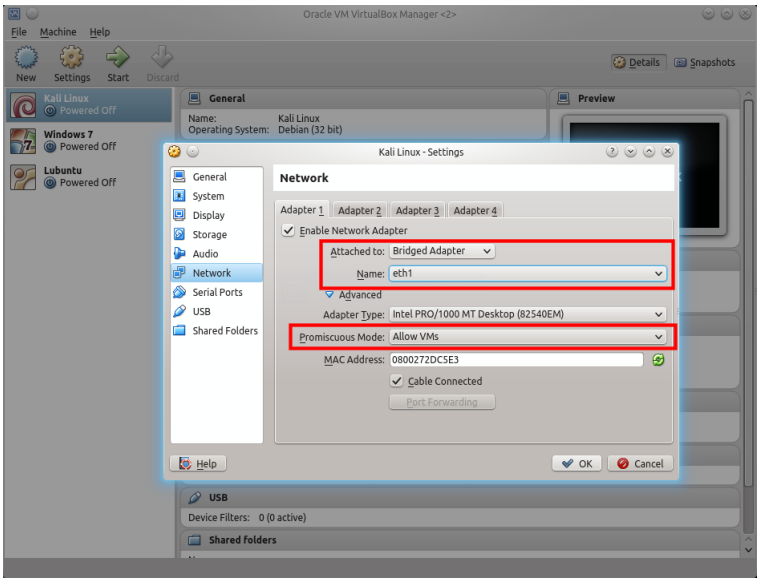


Figure 15: Changing your network configuration into "Bridged Adapter" mode.

After you have the network mode on the Kali Linux machine you will have to refresh your IP address:

```
# dhclient eth0 -r       % release your old IP address
# dhclient eth0          % get a new IP address
```

On your (non-clone) Windows machine make the same network changes as for Kali Linux. That is, change its network mode into "Bridged Adapter", however, *do not* enable "Promiscuous Mode" on it (it should remain at "Deny"). Verify that you have Internet access on both machines by opening a web browser and visit any web page. If the page loads successfully you can continue to the next section.

3.2 Sniffing Web traffic in Wireshark

We have prepared a very simple web page at the address `hacking.item.ntnu.no`. Here you can register and log in with a username and password (Figure 16). You will now use Kali Linux and Wireshark to sniff the user credentials of someone who logs on to this page.

First change your display filter to '`http`' in Wireshark. This will filter out everything except HTTP[5] traffic by changing the display filter to `http` traffic. Then visit

[5]Hypertext Transfer Protocol – the protocol used to transfer web pages.

`hacking.item.ntnu.no` with your Windows machine and create a user account on the page. After you have done this log in with your newly created username and password.

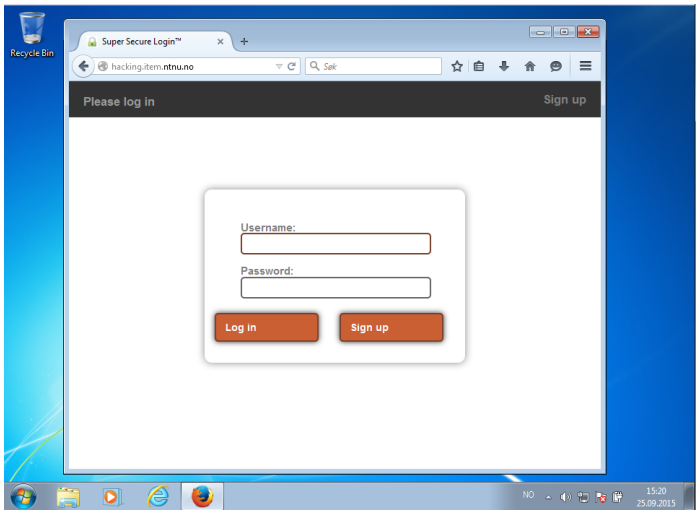


Figure 16: A simple web page requiring user log in.

What do you see in the Wireshark capture on your Kali Linux machine? Are you able to find the user's password?

3.2.1 Capturing images

There are several tools in Kali Linux that allows you to extract and display various parts of your capture data stream on-the-fly. For example, it is possible to display all the images a user visits in real-time using the utility `driftnet`. Run it as follows:

```
# driftnet -i eth0
```

In the browser of your Windows 7 machine try to visit `elg.no`. What do you observe on your Kali Linux machine?

Note: `driftnet` just shows a small subset of the data stream, while Wireshark of course captures *everything*. It is also possible to extract all images (in addition to *a lot* of other interesting stuff) from a saved capture in Wireshark by going to "File → Export Objects → HTTP".

3.3 Session hijacking

3.3.1 Session cookies

The web is inherently *stateless* which means that every connection that a user makes to a web page is seen as a completely new (and independent) connection by the web server. Unfortunately, without state it is not possible to create many of the features

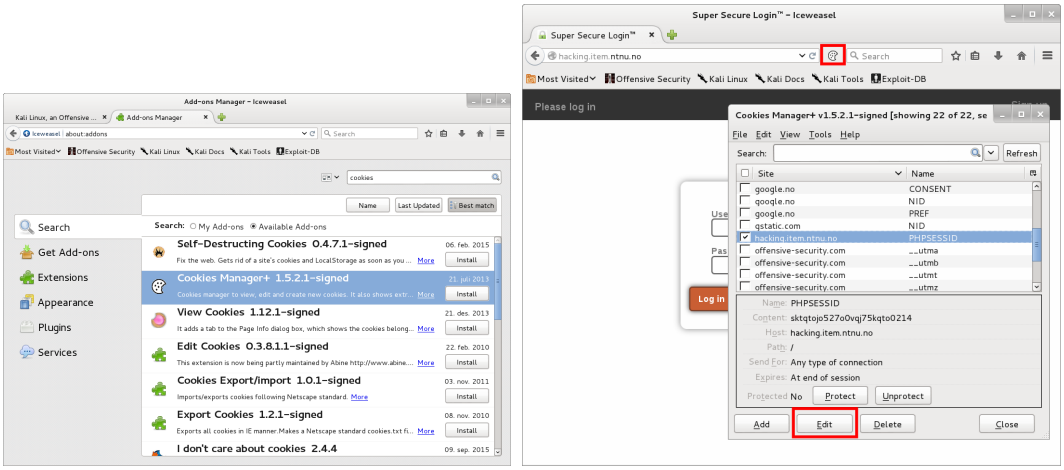
you would except from a modern web page. Thus, most web sites today employ some form of state handling. Typically, this is done by a combination of *server-side sessions* and *client-side cookies*. With server-side sessions, all relevant information pertaining to an ongoing connection is stored on the web server and the server also assigns a *session identifier* to it. With client-side cookies the same state is instead stored on the client in small information capsules called *cookies* which are transfered along with every HTTP request that your browser makes to the web server. Often a combination of the two is used, where the client-cookie merely contains the session identifier and the rest of the state is stored on the server. For example if a user is logged in to on a web site, its browser will send the session-cookie along with every request as he moves around on the site. On the server-side the server extracts the cookie from the request and checks whether it has stored any state for this particular session-ID, and if so, presents the page that a logged in user should be allowed to see.

3.3.2 Capturing and cloning session cookies

The `hacking.item.ntnu.no` web page uses session cookies to track its logged in users. Your task is to take over (i.e. *hijack*) an ongoing session by sniffing the session cookie with Wireshark. That is, you should bypass the login page without having to use the username and password of a legitimate user of the web site (pretend that you did not manage to capture this earlier).

Hint: For this task it may be useful to install a *cookie manager* in your browser on Kali Linux. A cookie-manager allows you to read, write, delete and modify the cookies stored in your browser. Several add-ons exists for this purpose.

In the *Iceweasel* browser inside Kali, click on "Tools → Add-ons" then type "cookies" in the search field and install the Add-on called "Cookies Manager" (after it is installed you will also have to drag its icon the Iceweasel menu (Figure 17b)). Note that any cookie manager that lets you add/edit cookies will do, it doesn't have to be this one. If you now browse to `hacking.item.ntnu.no` and click on the Cookies Manager icon and you will be able to edit your cookies for this site (Figure 17b).



(a) Installing a cookie manager Add-on. **(b)** Editing a cookie.

Figure 17: Using a cookie manager in Iceweasel.

After you have edited your cookie, refresh the site. What happens?

3.4 Web pages employing encryption: HTTPS

Like Telnet, standard HTTP sends all its traffic in the clear. To protect web traffic one have to use HTTP**S**[6] instead. Most major web pages today use HTTPS when they need to protect traffic that contains sensitive data.

In Wireshark change your display filter to `ssl`, then visit a page that employs HTTPS with your Windows machine. You can identify whether a site uses HTTPS by looking at the URL: it should start with "`https://`" (notice the 's'). Good candidate pages are `https://gmail.com`, `https://facebook.com` and `https://www.altinn.no`. Try to log in to any of these pages then check the Wireshark capture to see if your username and password appears anywhere (Hint: hopefully not!).

Questions

Q1. What is the difference between a *passive* and an *active* man-in-the-middle attack?

Q2.

a) Describe in detail how Ettercap were able to man-in-the-middle the connection in Section 2.6. In particular, include a simplified description (either in the form of

[6] HTTPS – HTTP Secure is basically normal HTTP, but instead of running (directly) over TCP, the web traffic is instead run inside of an SSL/TLS tunnel layered on top of TCP. SSL/TLS (Secure Socket Layer/Transport Layer Security) is a transport layer protocol that provides encryption and authentication of data.

a diagram or in words) of the packets that were sent to the switch and from the switch to the victims.

b) If you not only wanted to eavesdrop on the original Windows machine's communication with the Clone, but also wanted to man-in-the-middle its connection to the Internet, what IP and MAC address should have been Target 2 in Ettercap?

Q3. Some web pages, like `stackoverflow.com` or `amazon.com`, only deploys HTTPS on their *login pages*, but use plaintext HTTP for the rest of their site. Can you see any issues with this approach?

Hint/extra task: If you have an account on `stackoverflow.com` log in and use a cookie manager to inspect the cookies that the site has set – in particular, there should be a cookie with name `usr`. Now open an additional browser window, but this time in *incognito mode*. Make sure that the cookie manager is also available in incognito mode. While in incognito mode browse to the main page of `stackoverflow.com`, but do not log in. What happens if you now create a new cookie with the name `usr` and set its value equal to the cookie with same name in your *non*-incognito window?

Q4. How can we defend against the attacks in this lab?