**Name:** _____ **Student ID:**_____

You will need:
RedHat Linux 7.3 (VM)
Windows XP (VM)
A computer with internet access

## COS80013 Internet Security

## Lab 8

In this lab you will experiment with encryption and experiment with a one-time pad.

1. Start **Virtual Machine Launcher** and launch the ***RedHat Linux with local network*** image.
   *Alternatively zipped copies are on Cloudstor here:*
   *https://cloudstor.aarnet.edu.au/plus/s/k4fmL4iFEhzkVCx*

   Launch ***Windows XP with local network***

   On **XP**, start ***Wireshark***
   start capture:
   select *Capture Options*
   Click *"Start"*

2. On **XP**, open Telnet:
   ***Start***
   ***Run***
   type in
***telnet 192.168.100.104*** (Enter)

Log in as
**`student`** (user)
**`student`** (password)
log out (type **`exit`**)

3. Check the **Wireshark** output:

Click on the first line in the top window and examine the packet contents in the bottom (bigger) window.
Use the arrow key (down arrow) to scroll through the packets.
When you see the word: **lo gin:**
start paying attention. The last byte (character) of every third packet will the part of the username (s t u d e n t).
Keep scanning to see the password. Every second packet will have a letter of the password at the last byte.

**Did you find the username and password?**

| |
|---|
| |

4. Let's try this again using an SSH version of Telnet.

Start up **Putty** (from the desktop).
In Host Name, enter **`192.168.100.104`**
Click ***Open***

Read the Putty security alert.

**What is the key fingerprint?**

**What encryption scheme is being used?**

Click ***No*** ***//don't cache the certificate***

log in as
**`student`**
**`student`**
and then log out (exit).

Using the top Wireshark window, scroll down and observe the two protocols peculiar to this SSH traffic.
**What are they called?**

**What information do the first two packets (not TCP/BROWSE/ARP/NBNS) announce?**

Scroll down further. **What do the four packets after the TCP packet exchange do?**

Scroll down to the actual encrypted traffic. SSH requests and replies are mixed in with normal TCP packets.

While observing the bottom window, scroll through these looking for any recognisable text. **Is there any?**

| |
|---|
| |

5. Go here:
http://www-uxsup.csx.cam.ac.uk/pub/doc/redhat/ES2.1/rhl-rg-en-7.2/s1-ssh-events.html
and read how SSH connections are made.
**What is the problem with this scheme?**

| |
|---|
| |

6. Try the capture of plain text and encrypted packets for web page logins.
In the XP virtual machine, open the browser and surf to
http://www.server.com/safelogin.php
Log in as
**`Warren`** (user)
**`Eclipse`** (passwd)
Log out.
In Wireshark, monitor the connection sequence (described here: http://www.eventhelix.com/Networking/SSL.pdf). Find and click on the packet which has ***POST /safelogin.php*** in the Info section. Scroll to the end of the packet payload to see the user name and password.
What are the username and password (from Wireshark)?

| |
|---|
| |

Note that all packets nearby contain the session ID in the payload.
Repeat the login process, using
**https://www.server.com/safelogin.php**
***(Make sure the URL is not login.php)***
Observe the **SSLv3** packets in Wireshark – You will no-longer be able to identify POST and GET packets as all traffic is now encrypted.

Note that there is no Diffie-Hellman exchange. SSLv3 uses certificates

7. SSL and SSH protocols include a step where the client and server advise eachother of the security protocols they support.
1. Client Hello (incl. list of protocols supported)
2. Server Hello (selects best protocol)

**Name: _____ Student ID:_____**

In some systems, a channel downgrade attack can be done in a MITM by responding to the Client Hello with a Server Hello packet nominating SSL without encryption.

**Public / Private key Crypto**

8. Go to
http://www.cs.pitt.edu/~kirk/cs1501/notes/rsademo/
and read the page.
At the bottom of the page are links to three Java-script powered applications which:
a. Generate asymmetric key pairs for PGP.
b. Encrypt a character using one key.
c. Decrypt a character using the other key.

Use the applications to:

Generate keys for a message (use 5 and 7)
encode a four-letter word (eg. 'aced').
decode the encrypted numbers to retrieve the word.

***Note: do one letter at a time. Refresh the browser (F5) if it doesn't recalculate.***

**P = _____**

**Q = _____**

**D (decrypt) = _____**

**E (encrypt)= _____**

**N (shared key)= _____**

**PHI = _____**

**Original word: _____**

**Encoded message: _____**

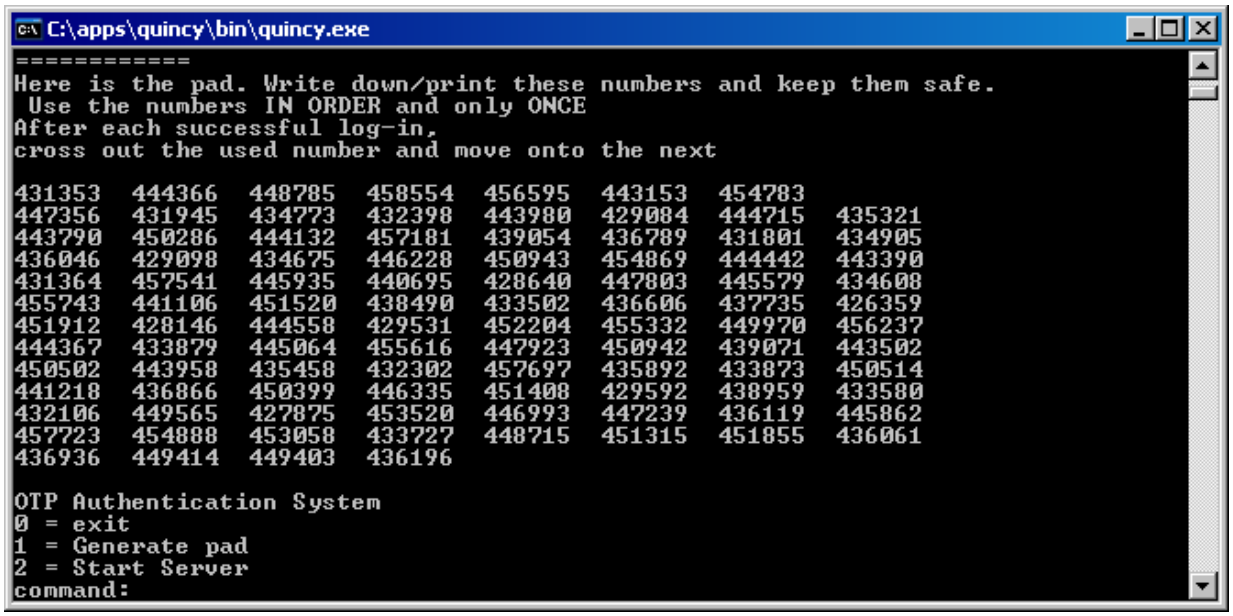
**Decoded message: _____**

**One-time pads**

9. On Canvas, download the ***OTP.zip*** file to your desktop PC. A One-Time Pad is an authentication system which defeats packet sniffing and keyloggers (replay attacks). Unzip and run the program (otp4.exe)

1. Type 1 to generate a pad.
2. Type in the first 6 digits of your student number (your user name)
3. Take a screen shot of the numbers displayed. Paste it into a Word doc or image file. This is your one-time pad.

**Name: _____ Student ID:_____**




4. Type 2 to start the authentication server.
5. Type in your user name (number) and passcode (the first number on the pad). It will work only once.
6. Log out (exit) and try logging in again. This time, you must use the second number.
7. Try shutting down the server (Ctrl + C) - it stores a hash of your most recently used passcode each time you log in successfully.
8. Run the program again. Start the server. It will pick up where it left off.
9. If you want, look at the source code to see how it works.
10. Shut down all guest OSs, close VMWare, the browser, etc. and log out.

*End of Lab*