**Internet Security – COS80013 Lab - 8 Report**

Internet Security – COS80013 Lab - 8 Report  
Student ID: 104837257  
Student Name: Arun Ragavendhar Arunachalam Palaniyappan  
Lab Name: COS80013 Lab 8 – Encryption, Telnet, SSH and One-Time PadsLab

Date: 08 /05/2025  
Tutor: Yasas Akurudda Liyanage Don

Sure, Arun. Here's your **Lab 8 report** rewritten with the **same content**, but now with **zero jargon**, **no fancy adjectives**, and in your preferred **plain, simple, human-like tone**, just like your Lab 7. Everything is kept straightforward and easy to follow.

**Internet Security – COS80013 Lab - 8 Report**  
**Student ID:** 104837257  
**Student Name:** Arun Ragavendhar Arunachalam Palaniyappan  
**Lab Name:** COS80013 Lab 8 – Encryption, Telnet, SSH and One-Time Pads  
**Lab Date:** 15 /05/2025  
**Tutor:** Yasas Akurudda Liyanage Don

**Title and Introduction**

This lab was about seeing how data moves across a network and what can be seen when it’s not protected. It showed how usernames, passwords, and session details can be picked up if encryption isn’t used. It also showed what happens when encryption is used and how it hides the data from anyone watching the traffic. Telnet, SSH, HTTP, and HTTPS were tested using virtual machines. Some hands-on tasks with RSA and one-time pad tools were also done to see how those systems work.

**Methodology**

The lab began with the Windows XP virtual machine and Wireshark. A Telnet connection was made to the RedHat machine using telnet 192.168.100.104, and the login was done with "student" as both username and password. Wireshark showed each character clearly, confirming that Telnet sends data without any protection.

Then, SSH was tested using Putty. The same login was done over SSH, and this time, Wireshark showed only encrypted packets. No readable data was found, proving that SSH hides login information.

Next, the HTTP login page http://www.server.com/safelogin.php was tested. The credentials "Warren" and "Eclipse" were used, and Wireshark captured them clearly along with the session ID. The HTTPS version of the same page was then used, and the traffic appeared encrypted, with no login info visible.

An RSA demo tool was used to generate keys using 5 and 7 as primes. A short word “aced” was encrypted using the public key and decrypted using the private key. The original word was recovered successfully, confirming how RSA works.

Lastly, a one-time pad program (otp4.exe) was run. A pad was created using the first six digits of the student ID. The first number worked for login, but reuse failed. Restarting the server continued from the next number. This showed that each code could be used only once, preventing reuse.

**Data Recording and Screenshots**

Wireshark showed the full Telnet login clearly, with each character showing up in order. When SSH was used, nothing readable appeared in the traffic. The HTTP login showed the username, password, and session ID inside the POST request. With HTTPS, the traffic was encrypted and the login details were not visible. The RSA task showed how a word could be turned into numbers and safely sent, then decrypted back. The one-time pad task showed that only the first unused number could be used to log in, and the system kept track of it.

**Discussion and Application of Learnings**

**Learning 1**  
It was observed that Telnet is not secure. When the login was done through Telnet, all the characters of the username and password were seen clearly in Wireshark. This showed that Telnet sends everything as plain text and can easily be read by anyone watching the network.

**Real-World Application**  
If any system still uses Telnet on a shared network, an attacker can capture usernames and passwords without needing any access to the system itself. This makes Telnet unsafe for any login or sensitive data.

**Learning 2**  
When the same login was done using SSH, none of the login details were seen in the Wireshark capture. The traffic was encrypted, and no readable data appeared. This showed that SSH protects the session from being watched.

**Real-World Application**  
Most servers now use SSH for remote access because it hides all the communication from network sniffers. If SSH is used, even if the network is being monitored, the attacker cannot see the login or commands.

**Learning 3**  
During the website login test, HTTP showed the username, password, and session ID in plain text. But when HTTPS was used, the same data was hidden. The encrypted traffic could not be read in Wireshark.

**Real-World Application**  
This shows why websites should always use HTTPS for forms, logins, and any personal data. Without it, anyone on the same network could steal passwords or take over sessions.

**Learning 4**  
The RSA test helped to understand how a message can be encrypted using one key and decrypted only with the matching key. The message "aced" was successfully sent and recovered.

**Real-World Application**  
RSA is used in secure websites, emails, and other online systems where the sender and receiver don’t share a common password. It allows safe communication by using public and private keys.

**Learning 5**  
The one-time pad test showed that a password could only be used once. If someone tried to reuse the same code, it failed. Even after restarting the server, it continued from the next number in the pad.

**Real-World Application**  
This is the basis for systems like token-based login or banking OTPs, where each code is valid only once. It prevents stolen or reused codes from being accepted, making attacks like replay harder.

**Limitations**

The lab used older systems and basic tools. Real systems use stronger encryption and updated protocols. Telnet is outdated, while SSH and HTTPS use more secure methods. RSA and OTP were simplified for learning. Still, the lab clearly showed how encryption protects data from being exposed.