Arab Academy for Science, Technology and Maritime Transport
College of Computing and Information Technology

Course	Networks Security (CCY3201)
Lecturer	Prof. Dr. Ayman Adel Abdel-Hamid
TAs	Abdelrahman Solyman

12th Project

- Install & deploy 2 virtual machines :
    1. Linux mint (client)
    2. Linux mint (server)

Part 1 (TLS) : **[11 Marks]**

Overview : Create a simple web application that uses HTTPS based on TLS that is initialized using OpenSSL, and capture the traffic using Wireshark.

1. Use OpenSSL to generate certificate for both the client and server. These two certificates are signed by root CA. Store each certificate inside the corresponding virtual machine, also state the best way to save such certificates and how the private keys are stored securely.
   (All steps involved in this phase should be shown in detail and explained briefly beside the commands and screenshot for each stage in this phase)
2. Build a simple TLS client/server application using the OpenSSL library web feature. Client will access the server through the built app on specific URL and get simple http response.

3. Capture the traffic between client/server from the moment the connection starts until server respond with the simple HTML page using Wireshark and save it as (tls_client_server_urIDs.pacap).

4. Identify the encrypted traffic and use session key to decrypt the traffic and show plain HTTP packets and save it as

(tls_decrypted_client_server_urIDs.pcap).

5. Fully understand and ability to explain each process from generating the certificate until receiving the HTML response. Including the cipher suites used their keys and the version of TLS and so on.
6. Take screenshots to prove your work.

Part 2 (SSH) : **[6 Marks]**

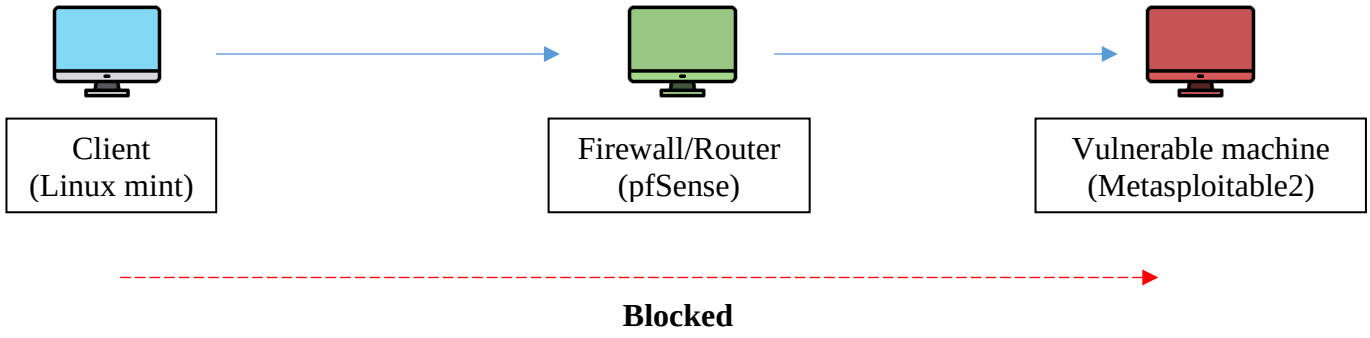
Overview : Install and configure SSH service to run on the server, generate server keys manually using ssh-keygen. Client should access the server using password, then should be able to access through public key generated. Capture traffic using Wireshark and generate SSH verbose log and explain it.

1. Install openssh-server and openssh-client, generate two key-pairs one for client virtual machine and another for your host machine using specific algorithms : ed25519, rsa.
2. Create a user named over your “ID” on server with password you choose.
3. Connect to the user from client with SSH using the password you created.
4. Generate key pairs on your client and copy the public key file to server through CLI, then connect using key-pairs.
5. Use PuTTY (or search for alternatives) from your host operating system and configure it to use the second generated public/private keys, login using the password first then use public key to login.
6. Disable password authentication and enable only key-pair authentication.
7. Capture traffic for SSH connection done using key-pairs and generate SSH verbose log for it and explain in detail.

- Deploy 2 more virtual machine :
    3. pfSense
    4. Metasploitable2

Bonus (Firewall IDS) : **[5 Marks]**

Overview : Install and configure firewall for example (pfSense + snort/Suricata for intrusion detection system - IDS) on virtual machine to set traffic rules and explore log alerts. Now user will try to access Metasploitable2 only through pfSense, block any direct access from client to Metasploitable2. Patch two of the vulnerabilities by simply blocking access to specific service/port. Download log alerts when client tries to run "Nmap" command on the vulnerable machine.



1. Deploy metasploitable2 and configure the network IP addresses and subnets such that the client can't have direct access to vulnerable machine.
2. Setup the pfSense to route traffic from client network to vulnerable machine through it.
3. Identify two vulnerable services log access to them, then apply firewall rules to block access to them.
   (All steps involved in this phase should be shown in detail and explained briefly beside the commands and screenshot for each stage in this phase)

Team size :

- Max 2.
- Each member of the team must participate in almost everything required in this project and expect to be discussed in all tasks individually.

Rubric [20 Marks + 5 Marks (Bonus)]:

- **[11 Marks]** Deploy simple HTTPS based on TLS built using OpenSSL
  - **[2 Marks]** Generate OpenSSL certificates, sign them, load and store certificates and private keys.
  - **[2 Marks]** Build a simple TLS client/server application.
  - **[1.5 Marks]** Capture TLS traffic using Wireshark.
  - **[1.5 Marks]** Decrypt TLS traffic captured using session key.
  - **[4 Marks]** Discussion
- **[6 Marks]** Install and configure SSH service with pair-keys to run on client and server.
  - **[1 Mark]** Installation and Key generation
  - **[2 Marks]** Login using pair-keys using CLI & PuTTY
  - **[2 Marks]** Capture traffic for SSH and generate SSH verbose log and explain it
  - **[1 Mark]** Discussion
- **[3 Marks]** Documentation well-organized for both parts including commands used brief explanation and step by step guide for your work with screenshot for each step.
- **Bonus : [5 Marks]**
  - **[1 Mark]** Setting up the environment as required
  - **[2 Marks]** Run scenario live during the discussion
  - **[2 Marks]** Prepare documentation and walkthrough for the lab with screenshots and steps to install and publish it to your Github repo.
- **To avoid losing marks follow these tips :**
  - Provide screenshot and commands for each different step you do
  - Screenshots should contain unique information related to you such as username of the OS, unique IP-addresses don't choose common IP-addresses.
  - Don't use LLMs (ChatGPT, DeepSeek ..etc) to generate your code, write code on your own even if it's not correct and try your best to make it functional.
  - Any type of plagiarism will be graded as **Zero**
  - Inability to distinguish your work from others or prove its authenticity will lead to losing marks, so try your best to make it unique and different.
  - Be prepared for discussion.

Deliverables:

- Submit the following on Classroom:
  - Team plan document specify what each member did.
  - Comprehensive, organized and well-formatted PDF report for the 2 parts of the project.
  - Text file contains IP addresses or subnets you used in each part.
  - Wireshark files for part 1
  - For bonus :
    - Text file contains IP addresses or subnets you used
    - IDS log file
    - Github repo link