NSR/AS Lab 3 – VPNs

Student Name: Tran Anh Thu Pham
Student ID: 103818400

TNE30009 – Network Security and Resilience
Swinburne University of Technology

103818400@student.swin.edu.au

***Abstract*— This report provides a comprehensive overview of Virtual Private Networks (VPNs), essential tools for secure communication over public networks. Beginning with an introduction to their VPN characteristics, capabilities, and various types including Remote access VPNs, Intranet VPNs, and Extranet VPNs. It highlights key considerations such as security, reliability, and manageability. This report delves into the fundamental concept of tunnelling within VPNs with its components and operations. Through detailed explanations, the role of tunnelling protocols such as IPSec, PPTP, and L2TP in encapsulating and encrypting data packets for secure transmission has been shown. Implementation using OpenVPN is demonstrated through shared key generation, transfer and the establishment of encrypted tunnels between virtual machines. Step-by-step instructions are provided along with insights into monitoring VPN traffic using Wireshark, emphasizing the encryption of data within the VPN tunnel. When comparing the monitored traffic from the tun1 and Ethernet interface, it becomes evident that encryption occurs solely on the tun1 interface after setting up the encrypted tunnel between two virtual machines.**

Keywords—VPN, OpenVPN, tunnel, shared key, encrypted traffic

I. Introduction to VPNs

A. VPN characteristics and capabilities

VPN (Virtual Private Network) is defined as a service that makes use of publicly available networking infrastructure to provide the features of a private network.
A VPN tunnel is an encrypted connection between a device and a VPN server. VPN Tunneling protocol has responsibilities for encapsulating data packets, IP packets and layer two frames for transmission. Different types of VPN protocols are IP Security (IPSec), Point-to-point tunneling protocol (PPTP), Layer 2 Tunneling Protocol (L2TP), etc.
There are three types of VPNs: Remote access VPNs, Intranet VPNs and Extranet VPNs.

General VPN requirements include VPN security, Availability, Reliability, Quality of Service, Compatibility and Manageability. To be more specific, security features such as firewalls, Network Address Translation (NAT), user and packet authentication, as well as encryption are necessary to be implemented to guarantee data transmission. Quality of Service (QoS) includes best effort, and relative, or absolute prioritization to optimise network performance. Availability and reliability ensure the Service Level Agreements (SLAs) from Internet Service Providers (ISPs). Redundant routing, access, and infrastructure from ISPs can enhance reliability. Compatibility requires IP gateways for non-IP traffic such as Frame Relay (FR) and IP tunnelling. Manageability is the ability to manage both remote and local sites.

VPN architectures consist of the following building blocks: VPN hardware (VPN servers, VPN clients, VPN routers, VPN concentrators, IP gateways, Virtual Private Routers, Virtual Private Trunking), VPN software(VPN server software, VPN client software, VPN management software), organisation's security infrastructure(firewalls, authentication systems, IPSec), service provider's security infrastructure(Reliability and availability through redundancy, Security through appropriate hardware and software, supports main tunnelling protocols and access networks), public networks and tunnels.

There are different ways to categorise VPN architectures based on their characteristics. Depending on who implements the VPN (ISP or subscriber), VPN architecture includes Dependent VPNs, Independent VPNs and Hybrid VPNs. Depending on where the VPN endpoints are, there are three types of VPN architecture namely Router to router, Firewall to firewall and Client initiated. Based on the network layer, there are Data Link layer (PPTP, L2TP), Network layer (IPSec) and Transport Layer (SSTP). VPN architectures are categorized based on their size, service and complexity e.g. class 0 for small organizations, class 4 for high speed access options.

VPN implementation brings its users a wide range of benefits in the digital world including low implementation and management costs, enhanced connectivity, robust security, efficient network capacity utilization, scalability, privacy protection and the ability to avoid geoblocking. However, dependence on the reliability of the Internet becomes one of the drawbacks of VPN.

Tunneling is the fundamental aspect of VPN involving encapsulating an entire data packet within the packet of other protocols. Tunneling components include Target network, Initiator node, Home Agent (HA), and Foreign Agent (FA). Tunnel operations include two phases respectively Initiation and Data transfer. In the first phase, the initiator sends the connection request to the Foreign Agent in the local network. Then the FA will authenticate the request. If the authentication is successful, the FA will send the request to the target network Home Agent. If the HA accepts the request, the FA will send the encrypted login ID and password. After the login is verified by the HA, the HA will send a Register Reply message and tunnel ID to the FA. Finally, when the FA receives the Register Reply from HA, the tunnel is established. During the second phase of VPN operations, the initiator begins sending data packets to the FA. Then the FA constructs a tunnel header and a header of a routable protocol, adding them to the data packet. After the tunnel header construction, the FA transmits the resulting routable encrypted packet to the HA using the provided tunnel ID. When the HA receives the encrypted information, the HA removes the tunnel header and routable protocol header. Finally, the original data is then

forwarded to the intended destination node. This process ensures secure and efficient data transmission between two nodes through the VPN tunnel.

A tunnelled packet includes the components illustrated in Fig 1 below.

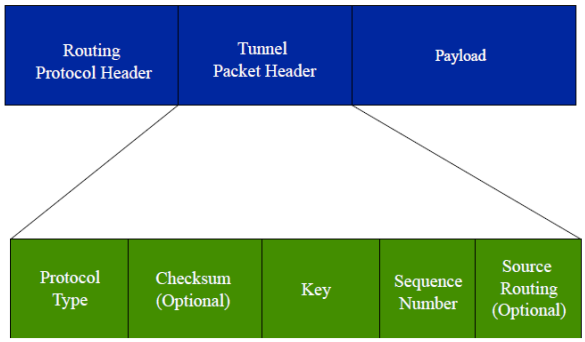

Fig. 1. Tunnelled packet format

A tunnel comprises three protocols: The carrier protocol, the encapsulating protocol, and the passenger protocol. To be more specific, the carrier protocol is responsible for routing tunnelled packets across the network, typically using IP. The encapsulating protocol adds necessary headers for secure transmission and the passenger protocol encapsulates the original data, often using PPP or IP.

There are two types of tunnels: Voluntary tunnels and compulsory tunnels. The key difference between these two types is their initiation and purpose. Voluntary tunnels are established end to end at the request of one of the endpoints. In contrast, compulsory tunnels are created and configured by an intermediate device. Voluntary tunnels serve the exclusive use of a single communication while compulsory tunnels are shared by multiple communications.

B. VPNs implementation in Organisational security policy

Implementing VPNs in organizational security policy safeguards sensitive data and ensures secure Internet communication. The organisations should point out their overall security objectives to specify who can use VPNs, for what purposes, and under what circumstances. The clear Organization's security policy is then defined to align with these requirements.

Access to VPNs is strictly controlled by implementing strong authentication mechanisms to restrict access to authorized people only. Some types of VPN authentication methods are Password-based authentication, multi-factor authentication (MFA), and Role-based access control (RBAC).

VPN connections are required to use strong encryption protocols such as AES (Advanced Encryption Standard) to protect data transmitted over that network. AES provides a high level of cryptographic strength of resistance against brute-force attacks. It operates by transforming plaintext data into ciphertext using a symmetric key algorithm, where the same secret key is used for both encryption and decryption processes. The symmetric key is securely exchanged between the VPN client and server during the establishment of the VPN tunnel.

Logging and monitoring mechanisms should be implemented to track VPN usage and detect any suspicious activities. Monitor VPN connections for signs of unauthorized access or potential security breaches. The monitoring strategy encompasses real-time surveillance and analysis of VPN traffic patterns, anomalous behaviours, and security events using intrusion detection systems (IDS).

All devices connecting to the VPN are required to adhere to minimum security standards. The deployment of endpoint protection solutions includes antivirus software, anti-malware scanners, and firewalls. Additionally, endpoint detection and response (EDR) should be employed to provide real-time visibility into endpoint activities, identify suspicious behaviours, and respond to advanced threats proactively.

To mitigate the risk of data leakage and insider threats, endpoint data protection measures such as data loss prevention (DLP) controls to safeguard sensitive information stored on endpoint devices. This includes encrypting data using full disk encryption (FDE) or file-level encryption to prevent unauthorized access to confidential data in the event of device theft or loss.

II. OpenVPN Behaviour

A. Premilinary

In the lab, a Linus software called OpenVPN is used to set up an encrypted VPN tunnel between two Virtual Machines (VMs). OpenVPN uses Transport Layer Security (TLS) for authentication and negotiation of an encryption algorithm. This process starts with the Ubuntu machine setups and host configuration. After downloading the two machines named VM1 and VM2, the IPv4 addresses of these machines must be identified by typing the command: *ifconfig* in the terminals.

The ethernet interface (enp0s3) should list the IPv4 address of the host after running the above command (Fig. 2). In this case, the IP address of VM1 is: *192.168.65.129* and VM2 is: *192.168.65.128*

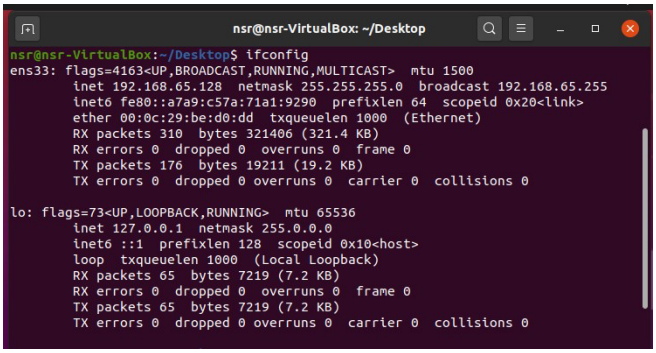

Fig. 2. IP address of VM2 in Ethernet interface section

To check the connectivity between two hosts, type the following command on the VM2 terminal to ping VM1 from VM2:

ping 192.168.65.129

It is similar to ping VM2 from VM1. If two hosts can ping each other, the results should appear as follows (Fig. 3).

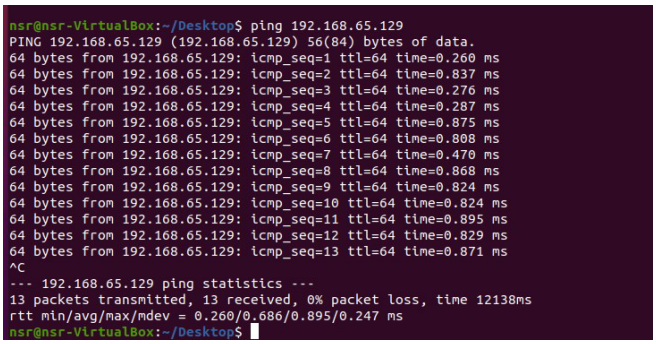

Fig. 3. Successful ping from VM2 to VM1

When needing to stop the ping, press ctrl-c.

B. *Shared password generation*

After verifying that the connection between two hosts is established, the shared password should be generated on VM2. In the command prompt Window, type:

Openvpn --genkey --secret mykey

The file called "mykey" will be created. To verify that the file is created or not, type the following command:

ls -l

This file should be listed as Fig. .

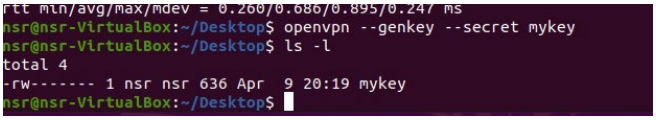

Fig. 4. Verification of the creation of mykey file on VM2

The file can also be seen on VM's Desktop (Fig. 5).

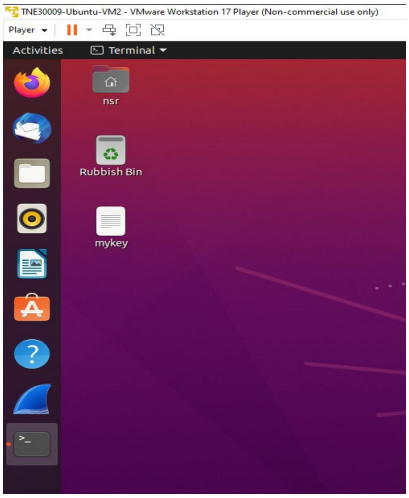

Fig. 5. Shared key file named mykey on VM's Desktop

To see the contents in the file, type the following command:

cat mykey

The contents of the file should be listed in Fig. 5.

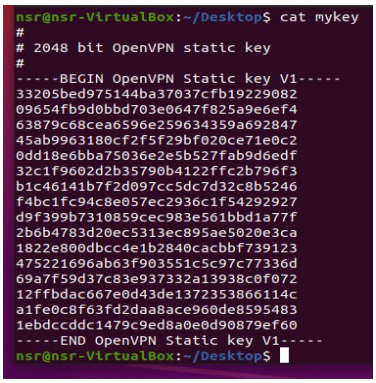

Fig. 6. mykey file contents listed

Now the shared key is created successfully on VM2.

C. *Shared key transferred to the other machine*

To transfer the shared key created in VM2 to VM1, Secure Copy (scp) is used on VM1 to copy the key from VM2. At the end of the process, both machines have the same file with the same contents.

The first step of this process is installing ssh which includes scp on each host. Type the following commands on both machines:

Sudo apt update

Sudo apt-get install openssh-server

The process of installing SSH is shown in Fig. 6.

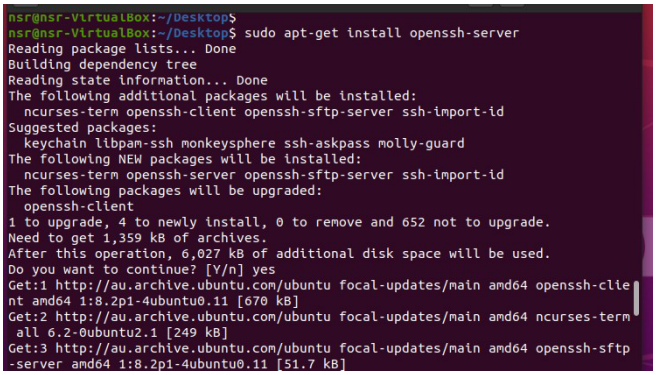

Fig. 7. ssh installation

The next step of this process is going back to the machine that generated the key previously, in this case, VM2. On VM2, type the following command (Fig. 8).

scp mykey nsr@192.168.65.129:Desktop/mykey

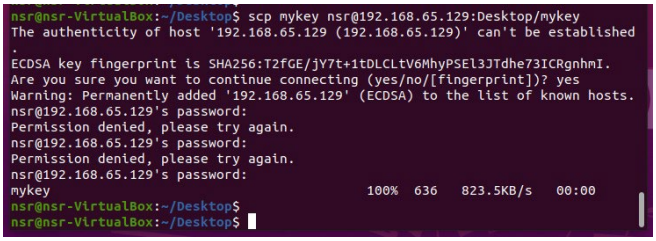

Fig. 8. Copy mykey file from VM2 to VM1

Note that the IP address in the above command is the IP address of the VM which copies the key from the VM that created the key before. In this case, the IP address of VM1.

After completing all the steps above, check if the shared key generated from VM2 is also copied to VM1 or not. To do this verification, type the following commands on VM1:

ls -l

cat mykey

The mykey file and its contents should be listed on the VM1 terminal similarly to Fig. 4 and Fig. 6.

D. Set up an encrypted tunnel

After the shared key are on both machines, the encrypted tunnel should be set up.

VM1 IP address: 192.168.65.129

VM2 IP address: 192.168.65.128

To set up the tunnel from VM1 to VM2 with the IP address of the tunnel endpoints 10.4.0.1 and 10.4.0.2 for VM1 and VM2 respectively, type the following command on VM1 (Fig. 9).

sudo openvpn --remote 192.168.65.128 --dev tun1 --ifconfig 10.4.0.1 10.4.0.2 --verb 5 --secret mykey

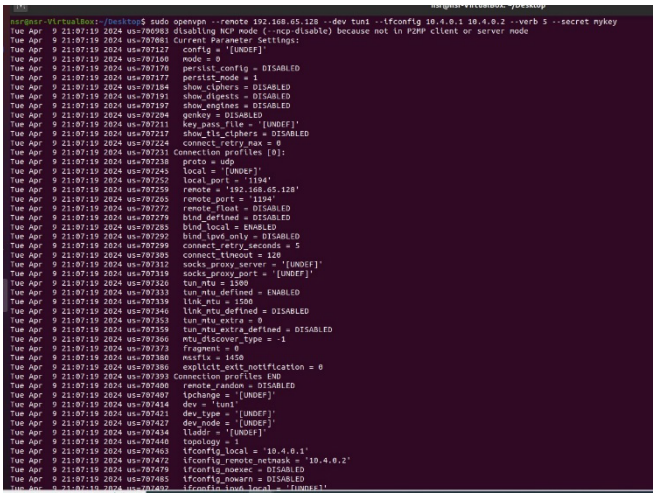

Fig. 9. Set up the encrypted tunnel from VM1 to VM2

This command sets up a tunnel called tun1 from VM1 to VM2. The IP addresses of the tunnel endpoints are 10.4.0.1 on VM1 and 10.4.0.2 on VM2. The remote address is the IP address of VM2 and the traffic going via the tunnel is encrypted using the key contained in the file “mykey”.

After that, the tunnel will be set up from VM2 back to VM1 with the remote address as the IP address of VM1, type the following command on VM2 (Fig. 10).

Sudo openvpn --remote 192.168.65.129 --dev tun1 --ifconfig 10.4.0.2 10.4.0.1 --verb 5 --secret mykey

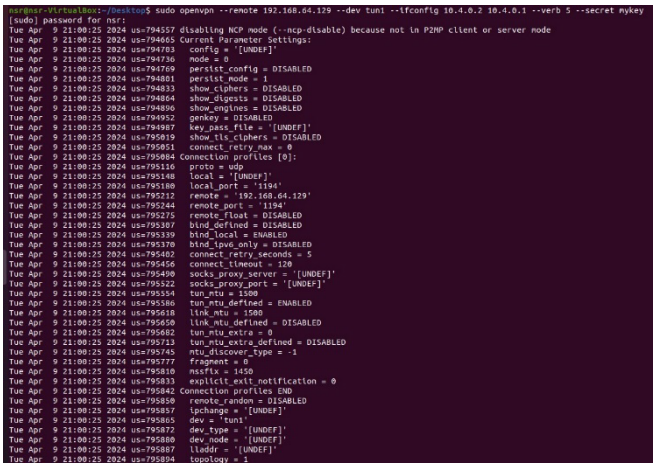

Fig. 10. Set up the encrypted tunnel from VM2 to VM1

To verify the encrypted bi-directional tunnel between both VMs, on both VMs, type the following command on the other terminals:

ifconfig

After pressing enter, a new interface called “tun1” will appear which is the encrypted tunnel created previously in terminals of VM1 (Fig. 11) and VM2 (Fig. 12).

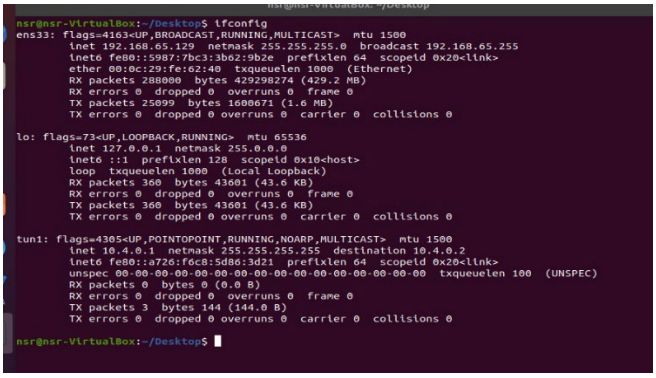

Fig. 11. tun1 interface appeared after setting up the encrypted tunnel on VM1

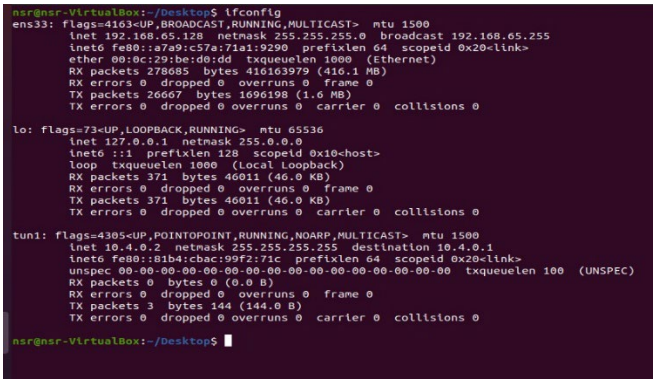

Fig. 12. tun1 interface appeared after setting up the encrypted tunnel on VM2

E. VPN tunnel monitoring on Wireshark

On VM2, ping the destination address of the tunnel, in this case, 10.4.0.1 (VM1 endpoint address).

ping 10.4.0.1

On Wireshark of VM2, the traffic from tun1 is captured when starting pinging the destination address of the tunnel (Fig. 13).

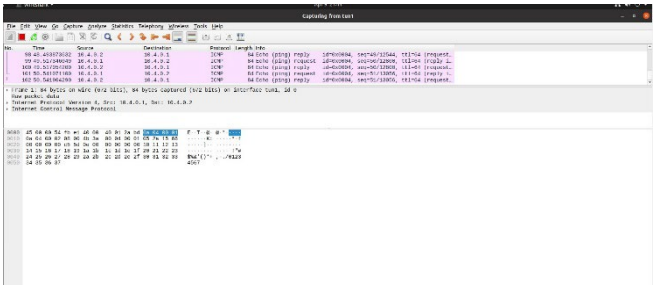

Fig. 13. Encrypted traffic captured from tun1 on VM2 Wireshark

On Wireshark of VM2, the traffic from the ethernet interface is captured when starting pinging the destination address of the tunnel (Fig. 14).

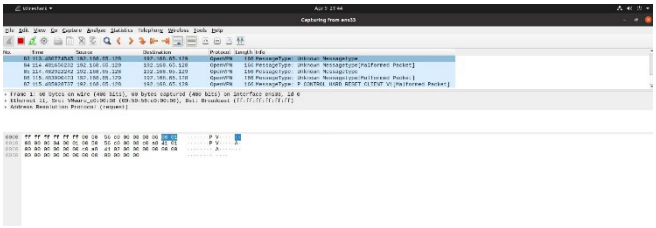

Fig. 14. Encrypted traffic captured from ens33 on VM2 Wireshark

After the telnet, monitoring the traffic captured from tun1 and ethernet interface respectively (Fig. 16 and Fig. 17).

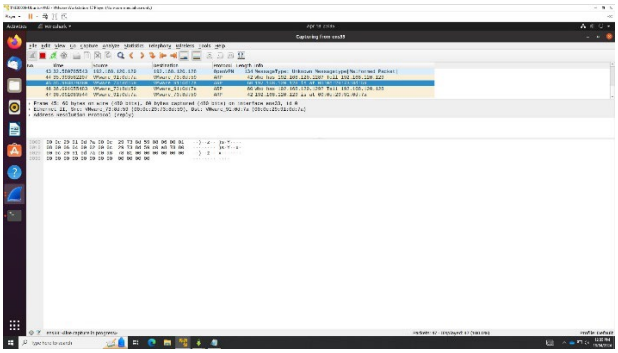

Fig. 16. Traffic captured from ethernet interface when using telnet

When monitoring the traffic from these two interfaces, tun1 and ens33, the traffic is only encrypted when captured from tun1. The traffic is encrypted when captured from tun1 because the VPN tunnel was established using OpenVPN with the same shared key on both machines. This VPN tunnel allows traffic to be encrypted when travelling through the VPN connection. The reason why the traffic is not encrypted when captured from the ethernet interface is because traffic on the ethernet interface is the original, unencrypted data before it enters the VPN tunnel. In another way, the VPN tunnel encrypts data before sending it over the physical network.

To telnet from VM2 to VM1, on VM2 type the command (Fig. 15):

telnet 10.4.0.1

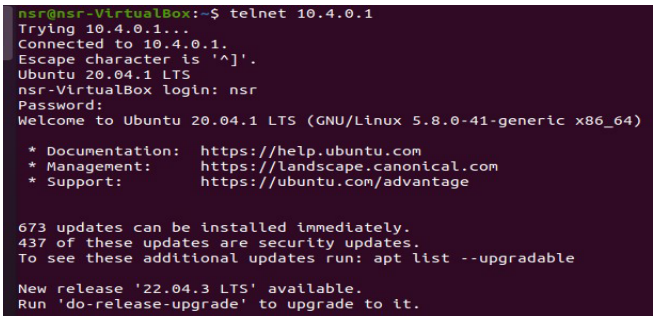

Fig. 15. Telnet from VM2 to VM1

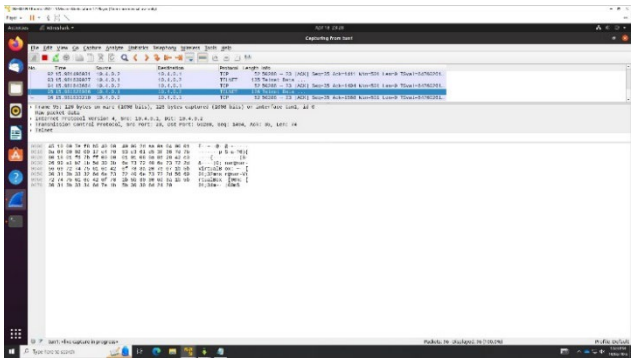

Fig. 17. Traffic captured from tun1 interface when using telnet

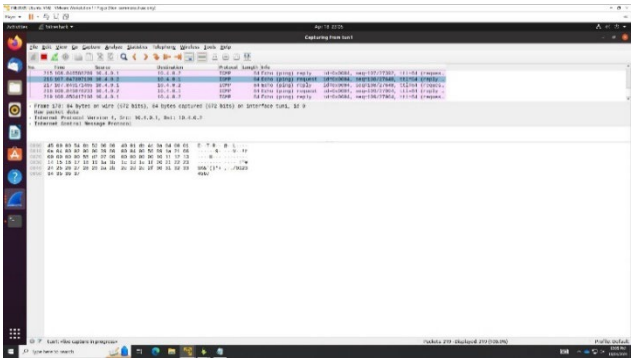

Fig. 18. Traffic captured from tun1 interface before using telnet

After using telnet, the tun1 interface is not encrypted, and traffic on the Ethernet interface is encrypted using the OpenVPN SSL protocol. The encryption status of the traffic on the tun1 interface depends on whether the application used for communication within the VPN tunnel has its encryption mechanism. In this case, telnet is used, which does not have encryption, the traffic would be fully decrypted on the tun1 interface.

At the end of the lab, the question " What is a VPN tunnel?" can be answered: It is a secure, encrypted communication pathway established over a public or untrusted network, enabling secure transmission of data between two endpoints. In this lab, the VPN tunnel was created using OpenVPN between VM1 (an IP address is 192.168.65.129 with remote address 10.4.0.1) and VM2 (an IP address is 192.168.65.128 with remote address 10.4.0.2).

III. Conclusion

In conclusion, this report has provided a comprehensive exploration of Virtual Private Networks (VPNs), and a description of their role in network security. By delving into VPNs ' characteristics, types, architectures of VPNs, their essential components and considerations involved in their implementation. From security and reliability to manageability and scalability, VPNs offer a robust solution for ensuring communication over untrusted networks.

Furthermore, VPN setup using OpenVPN has been demonstrated through shared key generation, transfer, and the establishment of encrypted tunnels between two virtual machines. Additionally, the monitoring of VPN traffic

using Wireshark has highlighted the encryption mechanism employed within VPN tunnels.

REFERENCES

[1]"Redirecting," login.microsoftonline.com. https://swinburne.instructure.com/courses/57020/pages/lectures-week-5?module_item_id=3844990 (accessed Apr. 28, 2024).

[2]"Redirecting," login.microsoftonline.com. https://swinburne.instructure.com/courses/57020/pages/laboratory-week-6?module_item_id=3844994 (accessed Apr. 28, 2024).

[3]"What Is Full-Disk Encryption? – Definition from TechTarget.com," WhatIs. https://www.techtarget.com/whatis/definition/full-disk-encryption-FDE#:~:text=Full%2Ddisk%20encryption%20(FDE)%20is%20a%20security%20method%20for

[4] A. Aarness, "EDR Security | What is Endpoint Detection and Response?," CrowdStrike, Feb. 06, 2023. https://www.crowdstrike.com/cybersecurity-101/endpoint-security/endpoint-detection-and-response-edr/

[5]"VPN Monitoring Methods | Junos OS | Juniper Networks," www.juniper.net. https://www.juniper.net/documentation/us/en/software/junos/vpn-ipsec/topics/topic-map/security-vpn-monitoring-methods.html (accessed Apr. 28, 2024).

[6] B. Team, "Mastering VPN Authentication: Best Practices for Enhanced Security," blog.bio-key.com. https://blog.bio-key.com/mastering-vpn-authentication-best-practices-for-enhanced-security