NETWORK SECURITY

Fundamental Principles and Practice

Shreya Wagley

wagleyshreya@gmail.com

Table of Contents

Introduction

Maintaining network security requires a thorough assessment of network vulnerabilities. This report will explore and experiment with UDP (User Datagram Protocol) port scanning, OS (Operating System) detection, and service detection, and attempt to illustrate their pivotal roles in identifying weaknesses, understanding system configurations, and ensuring the robustness of a network.

UDP port scanning involves probing a target computer to detect which UDP ports are open. Since UDP is a connectionless protocol, it poses unique challenges for security audits. This can often lead to costly oversights, leaving a network susceptible to vulnerabilities.
This report will explore the methods employed in UDP port scanning, delving into the packet-level intricacies of the process. Also, the differences in captured traffic between scanning open and closed UDP ports will be examined to shed light on how this technique uncovers potential security weaknesses.

Service detection, as the name states, is determining which services or applications run on a particular device. Having knowledge of the services running on a system is vital for security audits, as it aids in identifying potential vulnerabilities and backdoors. Service banners may not always provide accurate information about the services, and sometimes, they can be misleading. To ensure accuracy, service detection is performed.

In the same sense as Service detection, OS detection is simply determining the target computer's operating system. It empowers network administrators to make informed decisions regarding patch management, resource allocation, and overall security. This report will explore OS detection methods, with an emphasis on the packet-level details of the process. Limiting OS detection to specific open and closed TCP ports can help streamline the process, reducing network traffic while still yielding valuable information about the target system.

This report aims to provide a comprehensive understanding of UDP port scanning, OS detection, and service detection, signifying their importance in network security. By focusing on Nmap as a popular port scanner and OS and service detector, this report will showcase its capabilities in identifying potential security risks and optimizing network security measures.

UDP Scanning

Perform UDP scanning of a target computer to detect which UDP ports are open.

From a security auditing context, UDP scanning is quite important. Since UDP scanning is generally slower and more difficult than TCP it tends to be overlooked, however exploitable UDP services aren't uncommon. Thus, this neglect may potentially cause major vulnerabilities.

To perform UDP scanning of a target computer to detect which UPD ports are open, the command 'nmap -sU 192.168.0.102' can be used. The '-sU' option specifies a scan of UDP ports on the target, and the IP address is the IP address of the target. Figure 1 illustrates the result of performing this using 'Zenmap'; a GUI for nmap services.

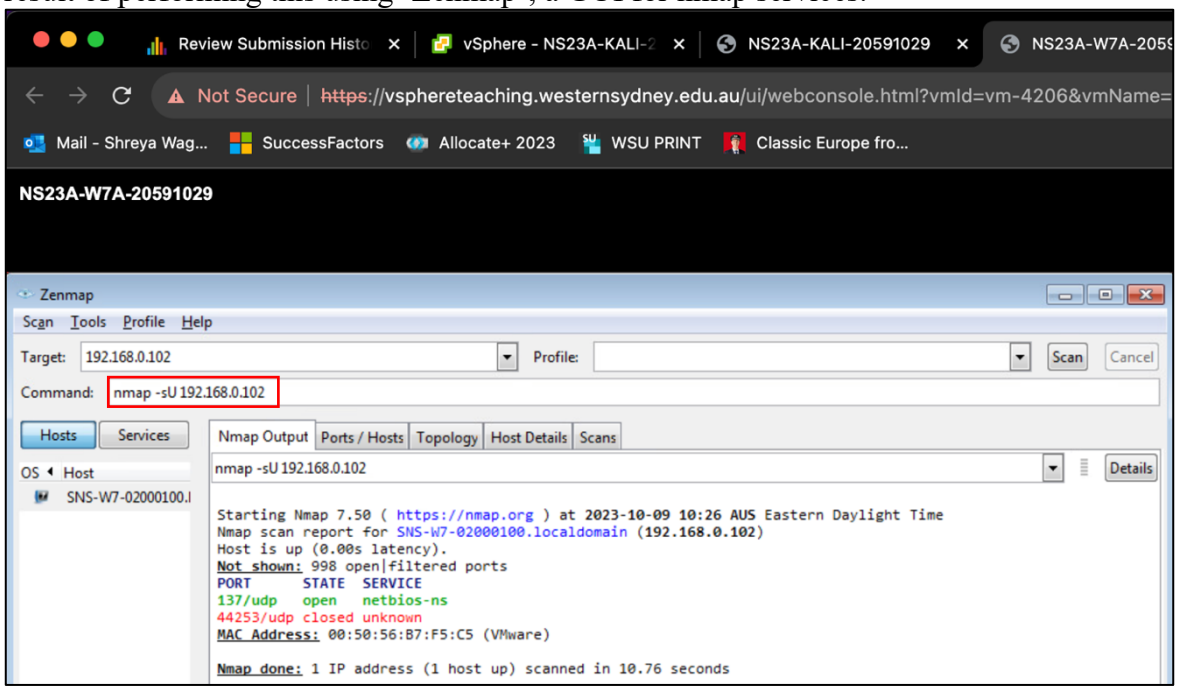
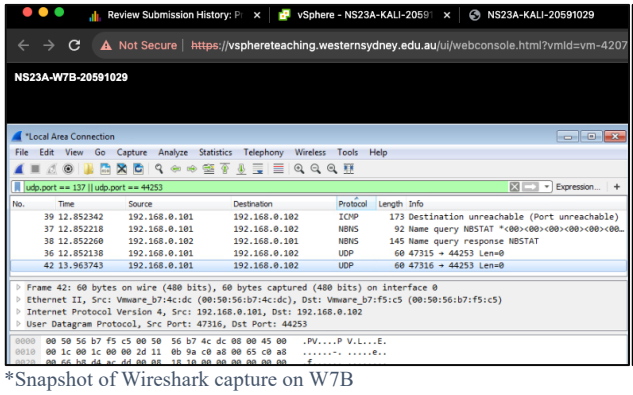


Figure 1

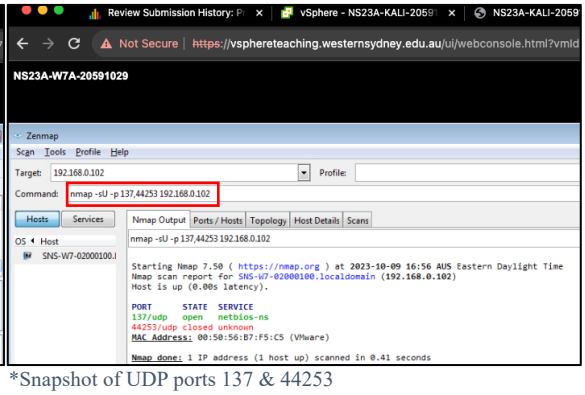
Pick an open UDP port and a closed UDP port to scan again, but this time, Wireshark running on the target computer, is used to capture the scanning traffic sent from the scanning computer to the target computer.

The purpose of using Wireshark to capture the scanning traffic in this scenario is gain insight into the actual actions that occur during a port scan, at a packet level. Since only two ports were produced in the preceding scan, 137 open & 44253 closed, these will be the chosen ports for this report.

To scan only these specific ports, the command 'nmap -sU -p 137,44253 192.138.0.102' can be used.



*Snapshot of Wireshark capture on W7B



*Snapshot of UDP ports 137 & 44253

Locate the UDP port scanning traffic and describe the traffic, especially the differences in the captured traffic between scanning the open and closed UDP ports.

To identify the differences in the captured traffic between scanning the open and closed UDP ports certain characteristics need to be examined. Typically, in open ports, a response such as "Destination Unreachable" or "ICMP Port Unreachable" can be observed. Indicating that the port is open, and the target responded with an error message. However, closed or filtered ports may often not even respond or respond with "ICMP Destination Unreachable". Also, searching for packets containing payload data specific to the service running of the port can also be helpful. On the contrary, closed ports typically don't contain payload data in the response packets. The description of the captured traffic is given in Table 1.

Table 1

No.	Description of traffic
39	Indicates port 137 is open, target responds with error message.
37	Attacker requests target's NetBIOS name
38	Target responds with its protocol statistics
36	Indicated attempt to reach port 44253, no response.
42	Indicated attempt to reach port 44253, no response.

To locate the relevant scanning traffic on Wireshark, the following display filter can be used 'udp.port == 137 || udp.port == 44253', this filter displays only the traffic associated to UDP ports 137 or 44253. Thus, in Figure 2, the top three packets are associated with port 137 (open) and the bottom two with port 44253 (closed).

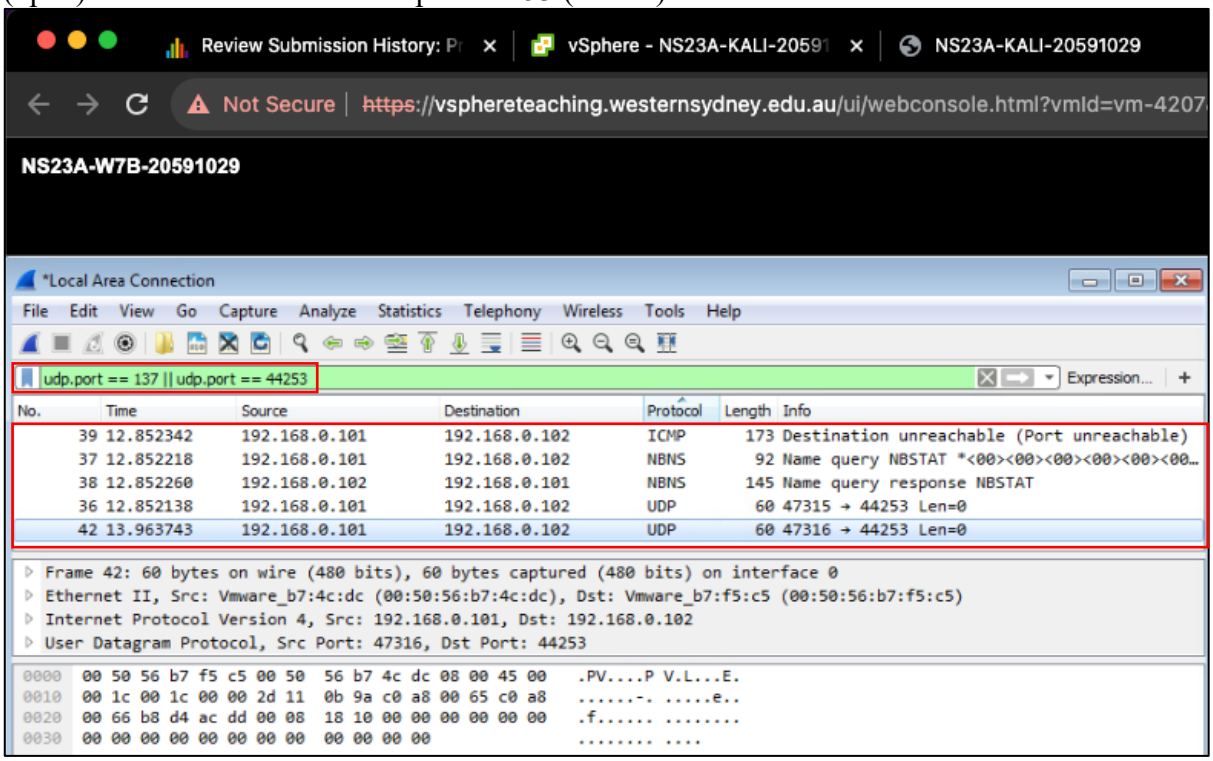


Figure 2

Service Detection

Perform TCP scanning of the target computer to detect which TCP ports are open.

From a service detection point of view, performing TCP scanning of the target computer is an effective way to identify which ports are open, potentially indicating the services or applications running on the target computer. This is valuable tool when conducting a security audit but can also be used maliciously to find commonly known vulnerabilities or backdoors that might exist in some services.

To perform a TCP scan the following command can be used in 'Zenmap': 'nmap 192.168.0.102'. In this case, no options are required in the command as nmap primarily performs TCP scans by default. However, if required the option '-sT' can be used to specify a scan of TCP ports on the target. The default scan will suffice for the scope of this report but in a real word scenario it might be wise to give a port range of 1 – 65535 (all possible TCP ports) using '-p 1-65535' as the default scan will only look at the top ports (most common 1000 TCP ports). This is demonstrated in Figure 3:

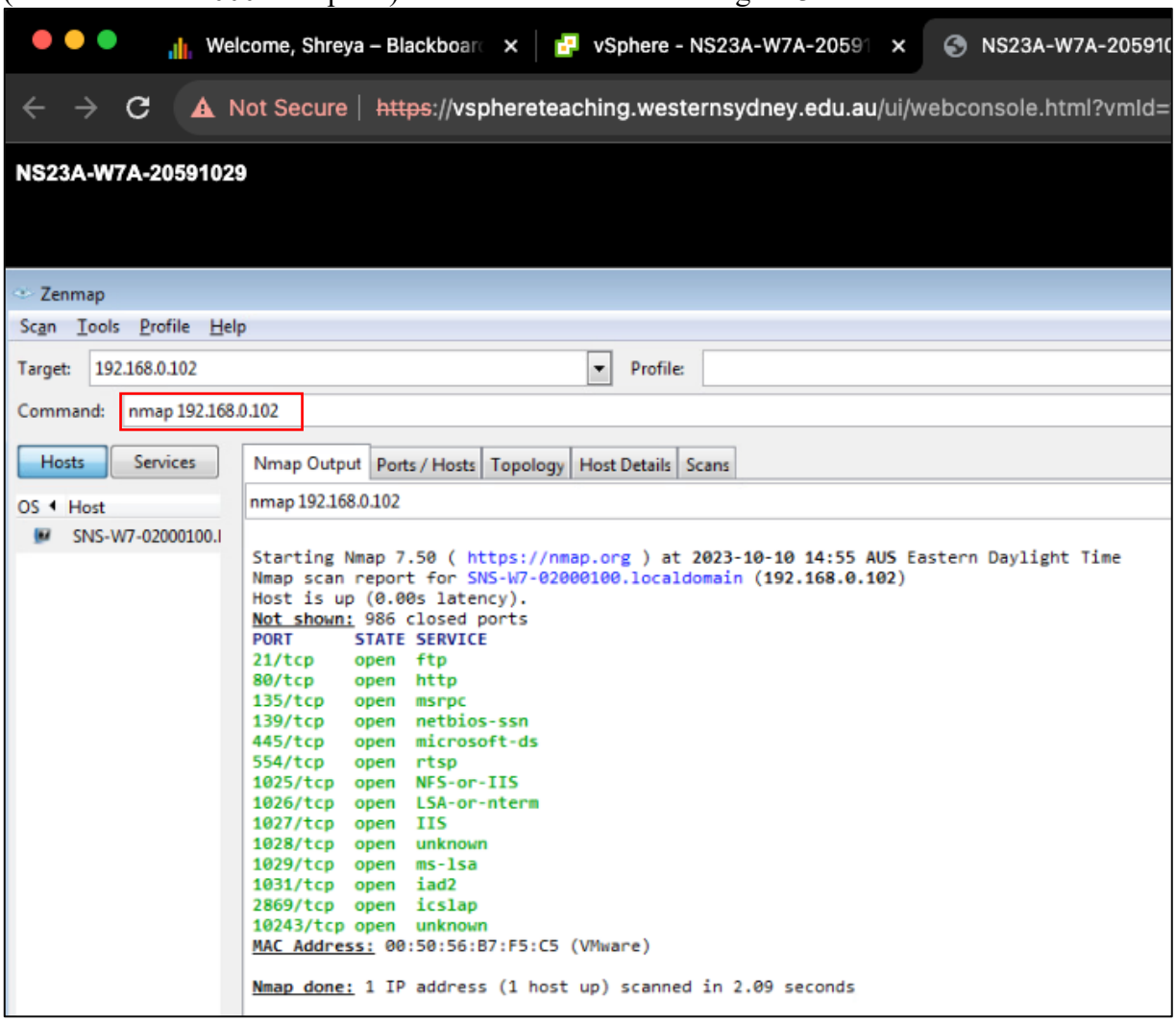


Figure 3

Pick an open TCP port to perform a service detection, and Wireshark is used to capture the scanning traffic sent from the scanning computer to the target computer.

Performing service detection is an integral part of network security as it allows one to identify, document and classify the services running on a particular device. The importance of this has been explained in the preceding section. By default,

This report will focus on port 135. In the snapshot above, the default nmap scan already provides the service running on this port. However, as a network security professional, one should not solely rely on this as people run services on strange ports and service banners can sometimes be bogus. Also it is very important to know which versions of a services is running, this helps in determining which potential exploits a server may be vulnerable to. Thus, for certainty, a service detection is performed. The command 'nmap -sV -p 135 192.168.0.102' can be used to accomplish this. The '-sV' option enables version detection and the '-p' allows the desired port to be specified. This is demonstrated in the snapshot below.

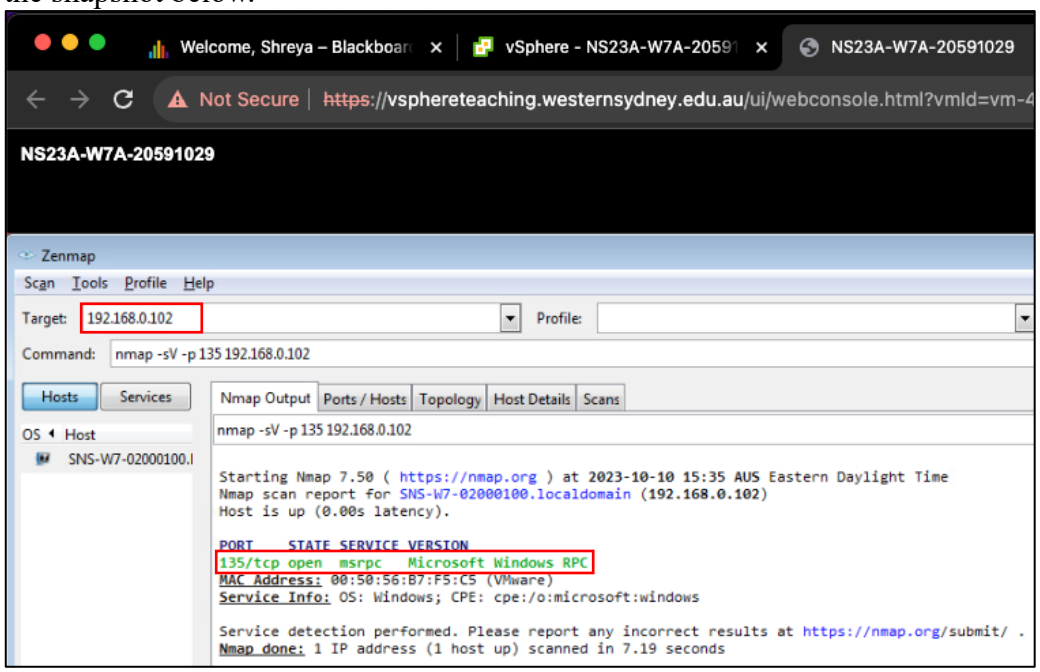


Figure 4

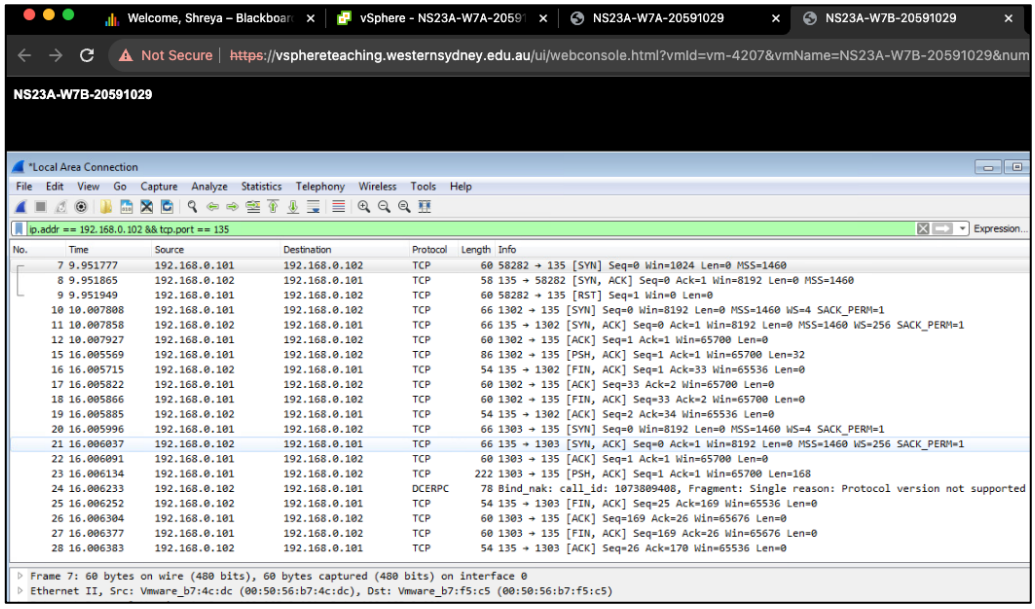


Figure 5

Locate the service detection traffic and describe the traffic. For example, for TCP 80, Nmap's Follow TCP may be used to reconstruct the scanning traffic for better presentation.

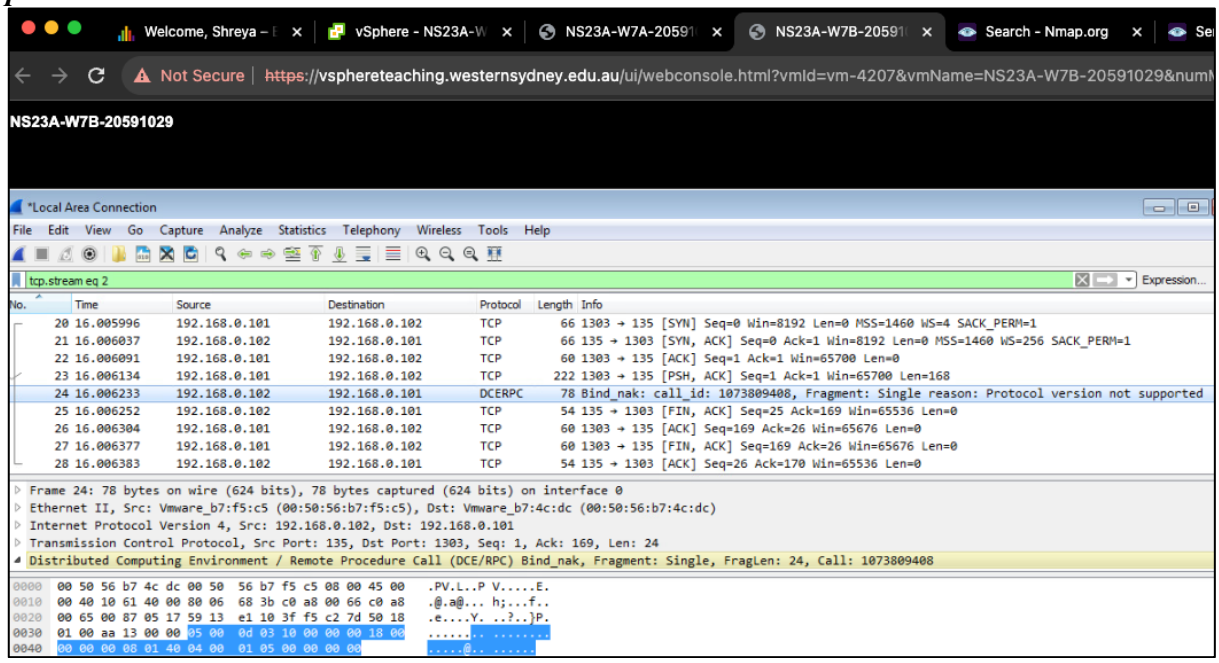


Figure 6

Figure 6 illustrates the traffic of a TCP connection between the scanning computer and the target's TCP port 135. First there is a successful TCP handshake, meaning an established TCP connection between the scanning computer and the target computer. Then the Push Acknowledgement packet may contain a request or query related to RPC service enumeration. Following this, the "bind_nak" response indicates that the RPC service on the target may not be available or accessible on the specified endpoint. Then the rest is usual connection termination procedure. This traffic pattern aligns with the typical behaviour of a scanning tool attempting to communicate with an RPC service to identify its presence and characteristics.

OS Detection

Perform an OS detection on the target computer, and Wireshark is used to capture the scanning traffic sent from the scanning computer to the target computer

Performing an OS detection is a fundamental step in network security as it provides crucial information about the target computer. This allows administrators to make informed and effective decisions regarding patch management, resource allocation and security.

To perform an OS detection on the target computer the following command can be used 'nmap -O 192.168.0.102'. The '-O' option enables OS detection. The output below shows that the target computer is running 'Microsoft Windows 7|2008|8.1', the '2008' is the model year and the '8.1' is the specific version. The output of this command using 'Zenmap' and the Wireshark capture are given in Figure 7 and Figure 8 respectively.

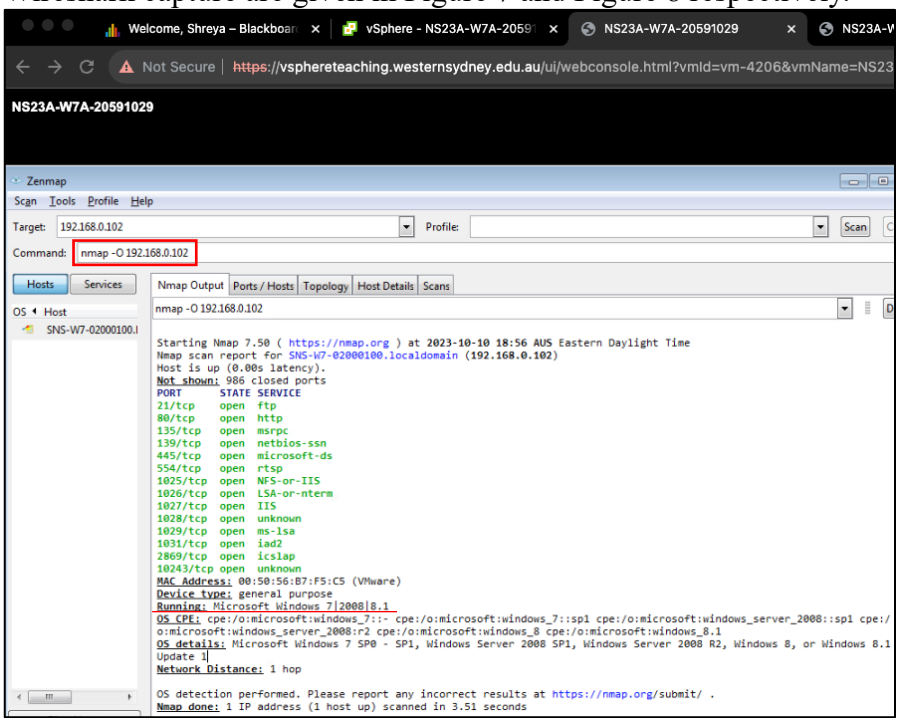


Figure 7

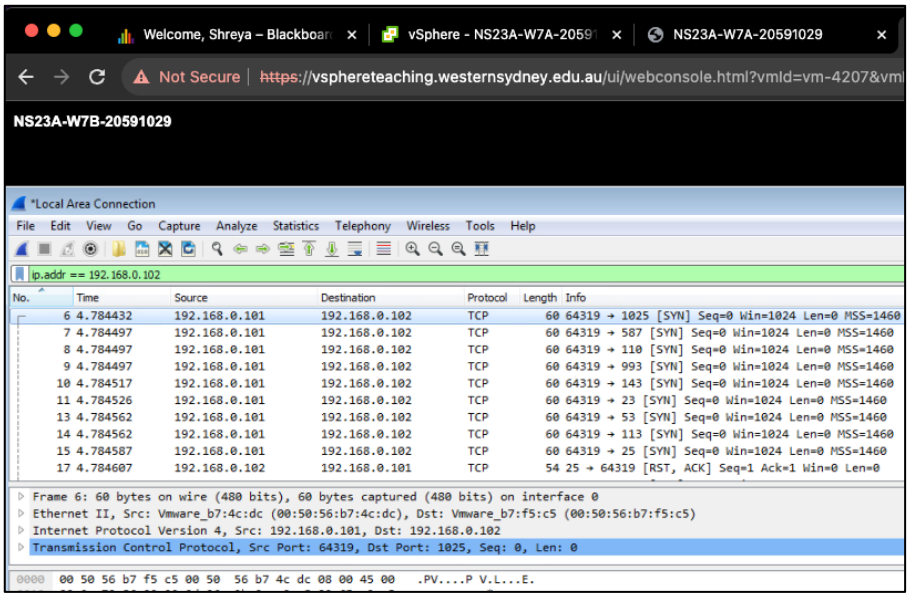


Figure 8

Locate the OS detection traffic and describe the traffic. To limit the amount of traffic, the OS detection may be limited to using one open and one closed TCP ports by using an Nmap option.

To limit the amount of traffic using one open and one closed TCP port the following command can be used 'nmap -O -p 135,9999 192.168.0.102'. It is known from the previous Service Detection section that TCP port 135 is open, the closed port can be any obscure port number; TCP port 9999 is used for this report. Figure 9 shows this.

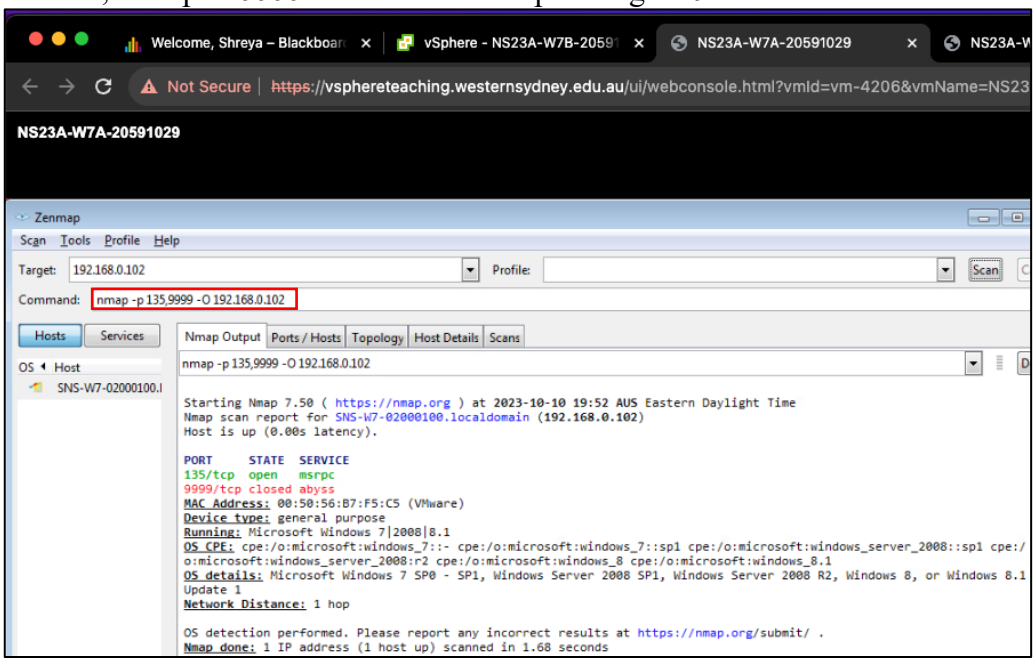


Figure 9

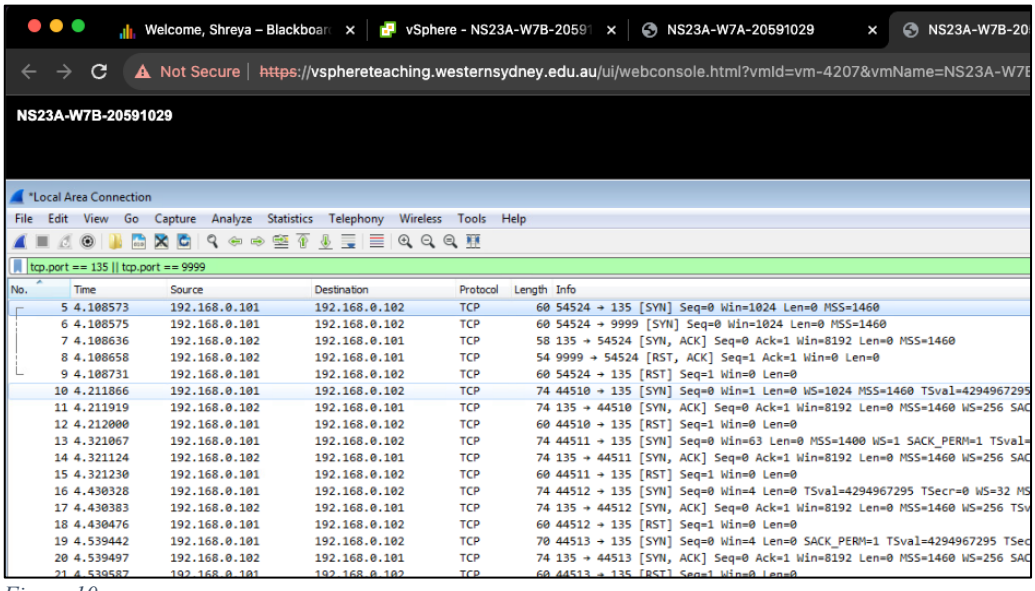


Figure 10

The Wireshark capture in Figure 10 shows the traffic that occurs during an OS detection scan. The first two TCP SYN packets are probe packets from the scanning computer to the target computer. TCP port 135 responds to this probe with a TCP SYN-ACK packet, acknowledging the request to make a connection. However, the target computer responds to the SYN packet sent to port 9999 with an RST indicating that it is a closed port, hence forth only the open TCP port 135 is involved. These numerous probing-and-acknowledging packets is what nmap uses to determine the target computer's OS.

Conclusion

In conclusion, this report has provided a practical exploration of UDP port scanning, service detection, and OS detection, emphasizing their crucial roles in bolstering network security. The practical application of these techniques demonstrated their significance in identifying vulnerabilities, comprehending system configurations, and ultimately fortifying network defences. The insights gained from UDP port scanning underscore its importance in uncovering potential oversights in connectionless protocols, while service detection emerged as a valuable tool for accurately identifying running services despite potential inaccuracies in service banners. OS detection, on the other hand, proved fundamental for making informed decisions regarding patch management and resource allocation. The hands-on validation of these methodologies using Nmap showcased their real-world applicability, reinforcing their role in optimizing network security measures.