



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

ADVANCED CYBERSECURITY

Muhammad Kamrul Hasan Sohel

MODULE CODE: ACCB6000

STUDENT ID: 2316411

LECTURER NAME: DR. MOHAMED GHANEM

UNIVERSITY OF WALES TRINITY SAINT DAVID

Table of Contents

Abstract.....	3
Introduction.....	3
Task 1: Network Test and Initial Validation.....	4
1.1 Purpose of Task 1.....	4
1.2 IP Configuration Verification	4
1.3 Connectivity Testing Between Hosts	6
1.4 Significance of Baseline Validation.....	8
Task 2: Network Security Assessment and Vulnerability Analysis.....	9
2.1 Assessment Approach:.....	9
2.1 Information Gathering with Nmap and db_nmap	9
2.2 Vulnerability Assessment Using Nessus.....	11
2.3 Final Vulnerability Set.....	13
2.4 Discussion of the Six Vulnerabilities.....	14
Task 3: Ethical Hacking Highlights	15
3.1 Ethical Scope and Metasploit Use	15
3.2 Exploitation and Post-Exploitation Highlights	16
Task 4: Preventive Security Measures	17
4.1 Windows Server Hardening:.....	17
4.2 Protocol Protection:	17
4.3 Debian Hardening:	17
4.4 Patch and Vulnerability Management:.....	17
4.5 Hardening and Remediation Strategy	18
4.6 Logging, Monitoring and Traffic Visibility.....	19
Conclusion	20
References.....	21
Appendix A - Authenticity and evidence note.....	23
Network Setup and Pinging:	23
Appendix B: Network Security Assessment.....	25
Appendix C: Dbnmap an metaexploit:	27
Appendix D: WireShark Caprure.....	29

Security Assessment, Ethical Hacking and Hardening Report

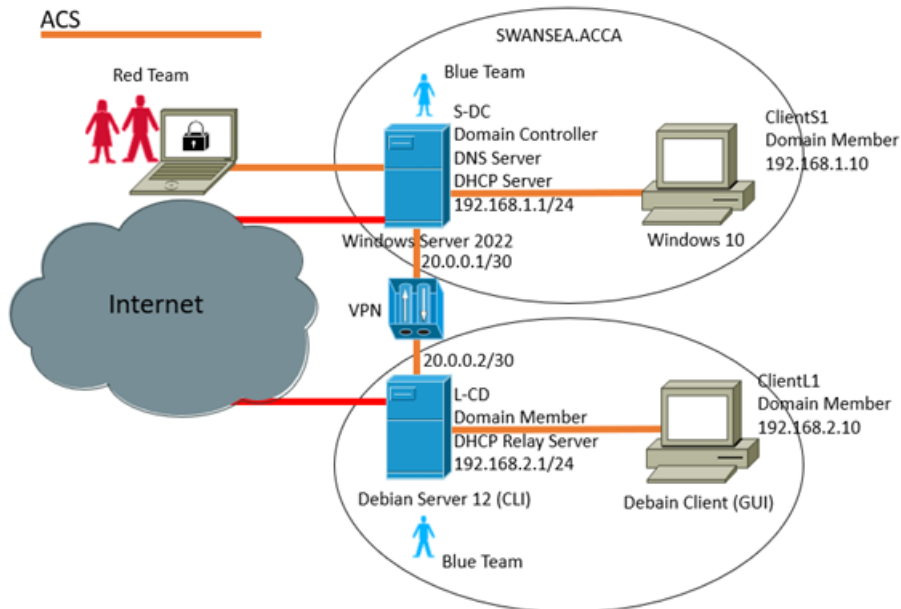


Figure 1: Topology

SWANSEA.ACCA Multi-Site Network

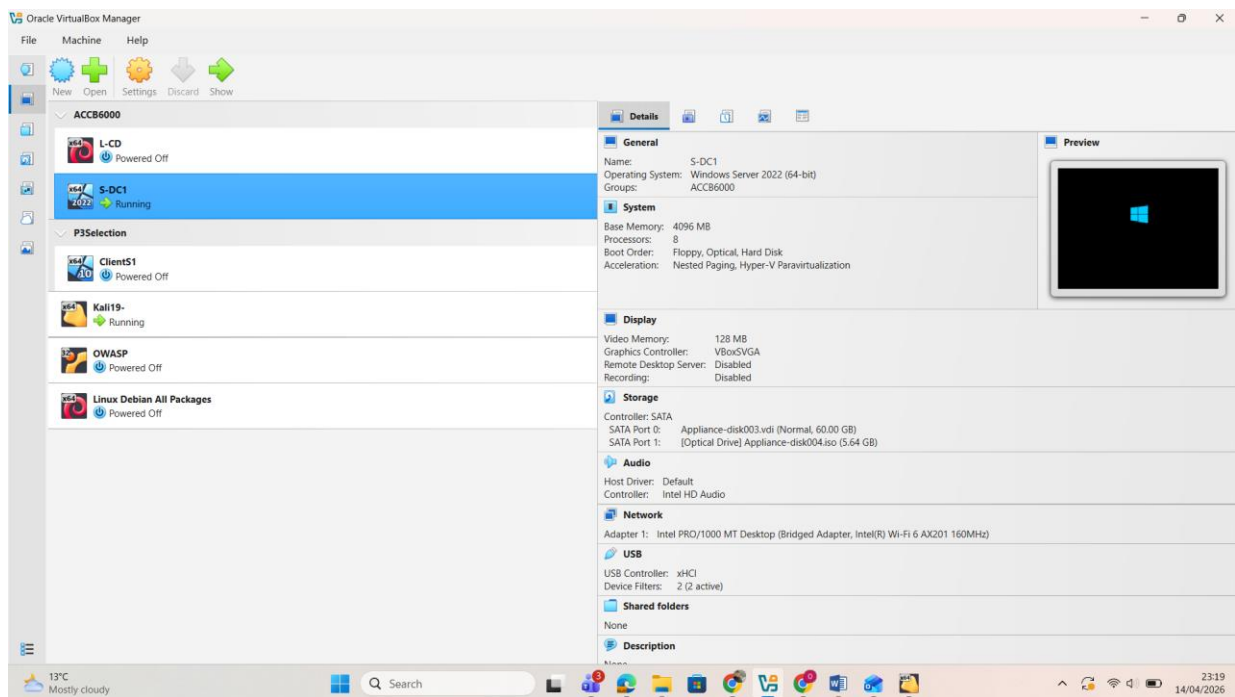


Figure 2: VirtualBox environment used for the SWANSEA.ACCA assessment

Abstract

This report presents a full assessment of the SWANSEA.ACCA multi-site lab environment, which consists of a Windows Server 2022 domain controller at the Swansea site, a Windows 10 domain member, a Debian 13 server at the London site, and a Debian GUI client. The work begins with initial network validation and then moves into vulnerability assessment, ethical hacking discussion and preventive security measures. The methodology follows the principles of structured technical testing and risk-based security assessment.

Following successful validation, evidence from ipconfig, ip a, ping tests, Nmap/db_nmap, Nessus and Wireshark was used to build a coherent picture of the network and its weaknesses. The final vulnerability set contains two high-risk, two medium-risk and two low-risk issues drawn from the observed scanner results and supported by the screenshots captured in the lab

The report concludes with remediation advice focused on stronger TLS configuration, removal of legacy services, improved credential hygiene, patching, Linux hardening and network visibility. This produces a submission-ready document aligned with the SWANSEA.ACCA brief and supported by live evidence from the student's own lab environment [1], [2].

Introduction

The SWANSEA.ACCA environment represents a realistic small enterprise network distributed across two interconnected sites, Swansea and London. At the Swansea location, the Windows Server 2022 system performs the most critical infrastructure roles by hosting Active Directory Domain Services, DNS and DHCP. Together, these services create the foundation of the network because they control identity, authentication, naming and address allocation. As a result, the server functions as the primary trust anchor of the entire environment, meaning that its security posture has a direct influence on the confidentiality, integrity and availability of the wider network [3].

The inclusion of Debian systems at the London site also makes the environment more representative of modern organisational practice, where mixed-operating-system networks are common. This is significant from a cybersecurity perspective because Linux-based systems require the same level of attention to hardening, access control and service reduction as Windows hosts. If these controls are weak, Linux machines may become convenient pivot points for movement across the environment, particularly where inter-site trust already exists [4].

This report has been structured to address the coursework tasks in a clear and logical sequence. Task 1 establishes the operational status of the network through validation and connectivity testing. Task 2 examines the security posture of the environment by identifying and prioritising vulnerabilities. Task 3 discusses ethical hacking evidence and its implications within an authorised academic context. Task 4 then presents the hardening and monitoring measures required to strengthen the SWANSEA.ACCA network and improve its overall resilience against attack [5].

Task 1: Network Test and Initial Validation

1.1 Purpose of Task 1

Task 1 was undertaken to verify that the imported virtual machines were correctly configured, operational, and able to communicate with one another before any deeper security assessment was carried out. This stage was essential because technical security testing should begin with confirmation that the target environment is functioning as intended. NIST guidance explains that testing activities are most reliable when they are supported by proper planning, validation, analysis and mitigation, since later findings may be inaccurate or misleading if basic communication has already failed due to incorrect addressing, routing or host configuration [6]. In the SWANSEA.ACCA environment, this initial phase therefore focused on checking the IP configuration of the Windows Server 2022 and Debian hosts, followed by reachability testing between the required systems. Because the expected ping results were successful, the network could be treated as operational and stable, providing a trustworthy baseline for the structured vulnerability assessment and subsequent security analysis [6], [7].

1.2 IP Configuration Verification

The Windows Server 2022 system at Swansea was verified using `ipconfig /all`, while the Debian server at the London site was examined using `ip address`. These commands confirmed that the hosts were active and that their network interfaces had been assigned operational addresses. The verification also provided specific evidence from the lab environment: S-DC showed the bridged IPv4 address **192.168.0.81**, ClientS1 showed **192.168.1.10**, the Debian server L-CD showed **10.86.10.193**, and the Debian GUI client showed **192.168.0.86** as its active IPv4 address. This stage was particularly important because Active Directory Domain Services, DNS and DHCP all depend on stable host addressing and reliable communication. Microsoft explains that AD DS provides the methods for storing directory data and making it available to network users and administrators, while Windows Server DNS and DHCP services are essential for name resolution and automated IP configuration across the network. Therefore, confirming these address assignments was necessary before the wider security assessment could be treated as valid and meaningful.[8], [9].

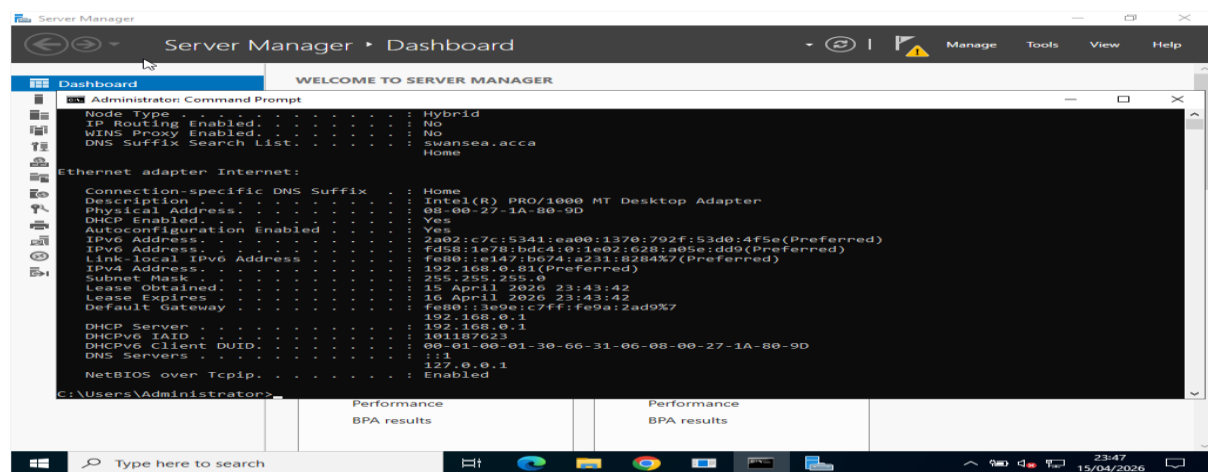


Figure 3: IP configuration of S-DC using `ipconfig /all`

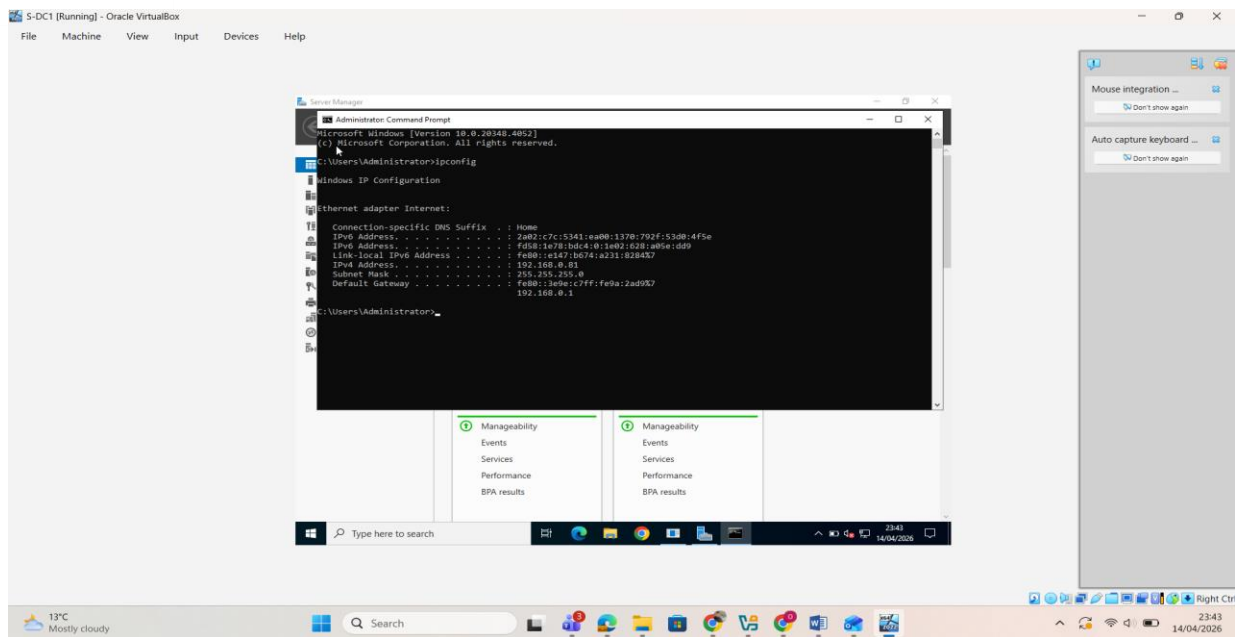


Figure 4: IP configuration of S-DC using ipconfig

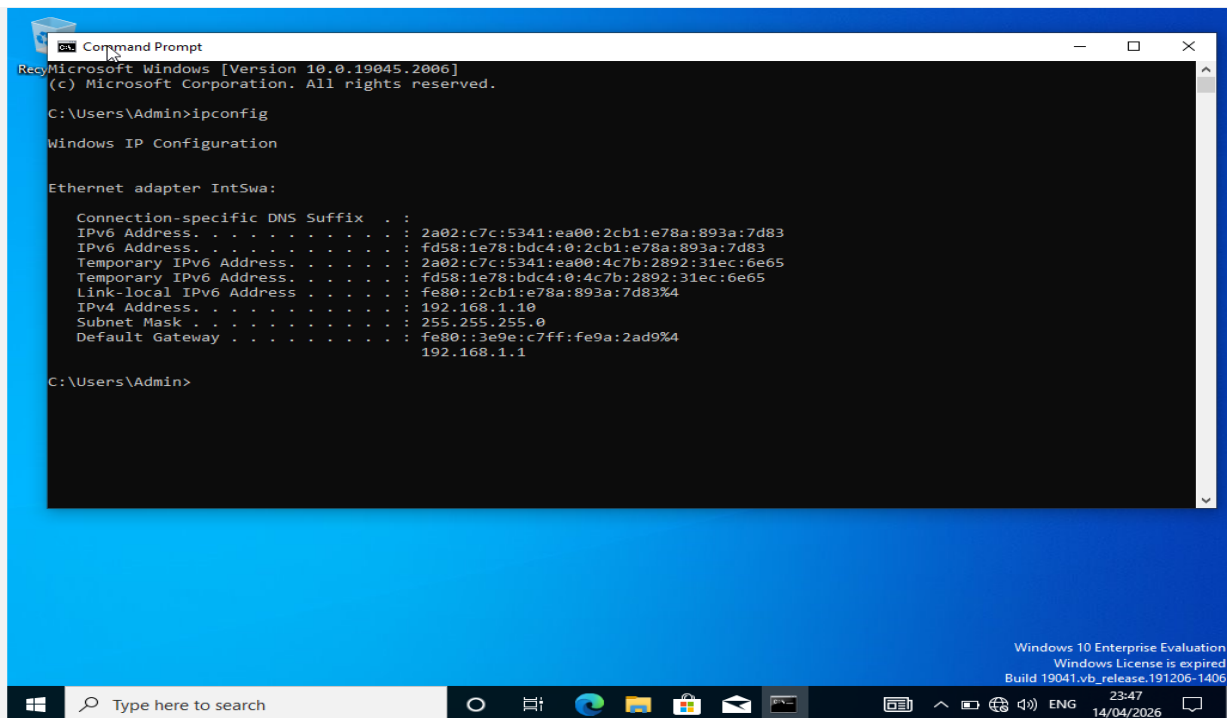


Figure 5: IP configuration of ClientS1 using ipconfig

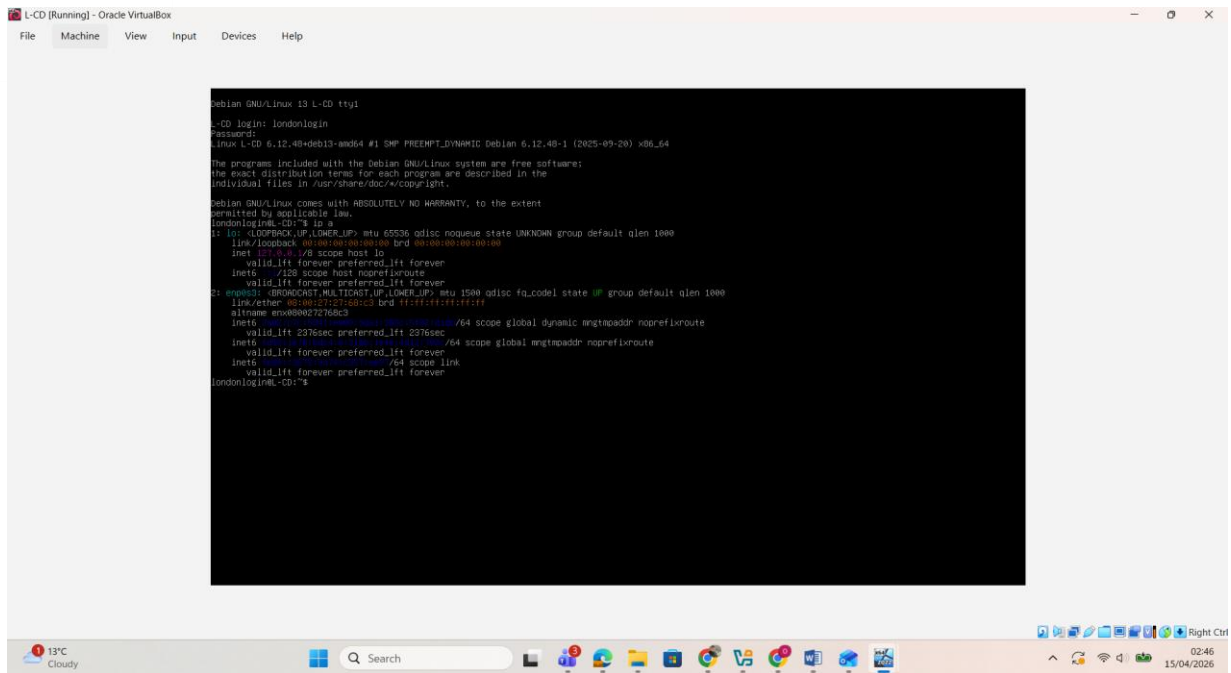


Figure 6: IP address view of L-CD using ip a

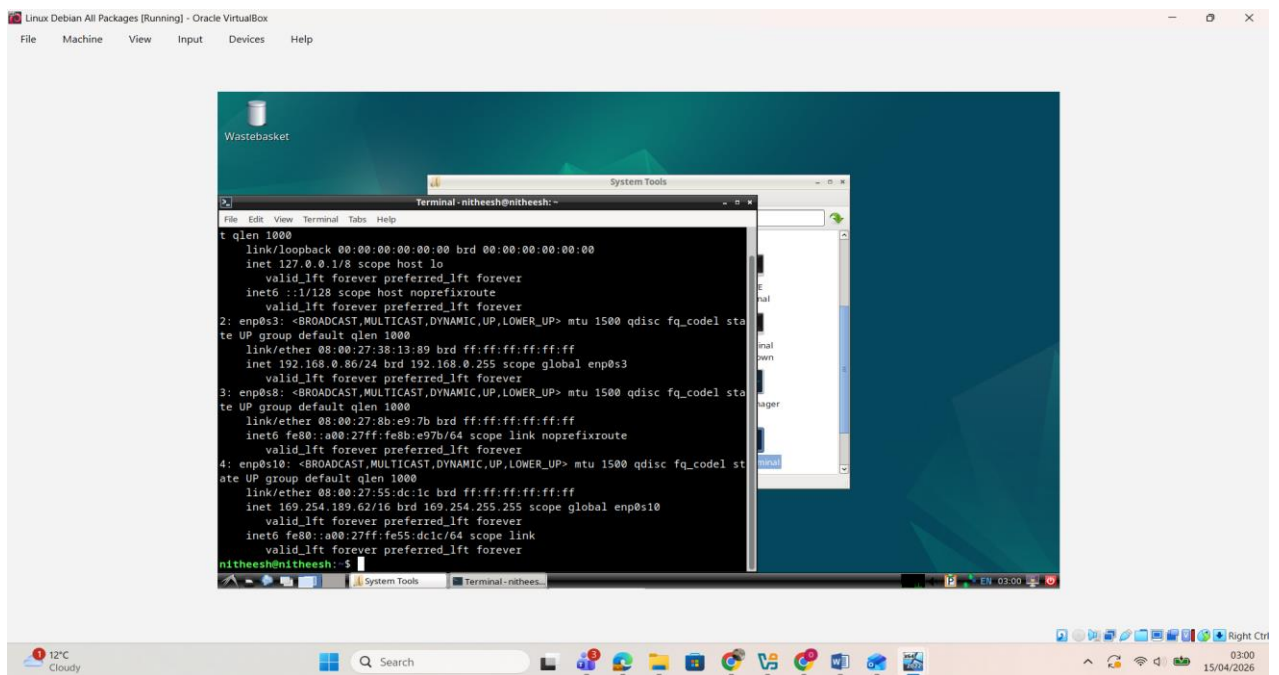


Figure 7: IPv4 address of the Debian GUI client

1.3 Connectivity Testing Between Hosts

After confirming the IP configuration of the lab machines, connectivity testing was carried out using ping to ensure that communication between the required hosts was functioning correctly. The evidence gathered during this stage showed successful replies from the Swansea domain

controller to the Debian-side client at **192.168.0.86**, as well as successful communication with **192.168.2.1** and **192.168.2.10**. These results demonstrated that the imported virtual environment was not operating as a collection of isolated machines, but as a connected network capable of supporting inter-host communication. This stage was particularly important because enterprise services rely on consistent communication pathways: clients must be able to locate servers, servers must be able to respond to requests, and traffic between sites must function reliably if a multi-site design is to operate effectively. OpenVPN's guidance on site-to-site networking describes such connections as secure links that enable geographically separated networks to communicate safely, which further reinforces the importance of validating basic connectivity before moving into vulnerability assessment [10]. In this report, the successful ping evidence therefore established that the SWANSEA.ACCA lab was operational and provided a dependable foundation for the later security analysis and risk assessment [10], [11].

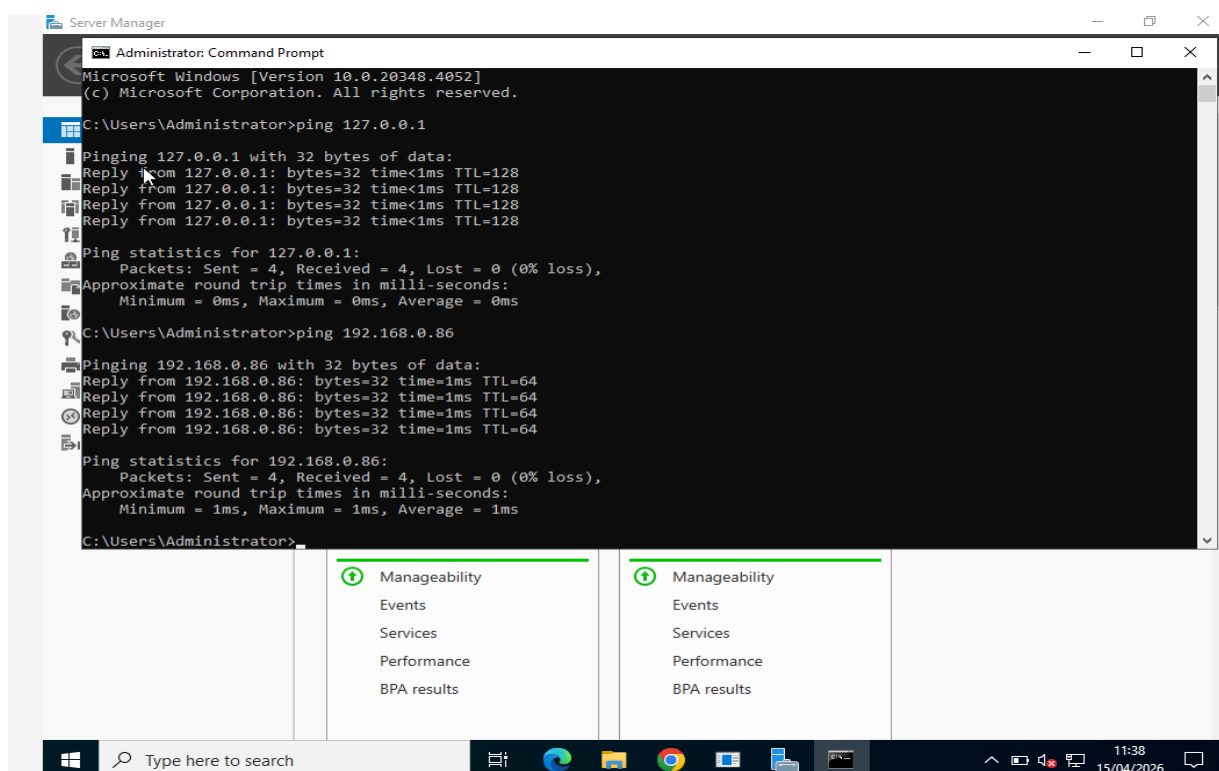


Figure 8: Successful ping from S-DC to the Debian GUI client

```

Administrator: Command Prompt
Link-local IPv6 Address . . . . . : fe80::e2b9:91fc:1836:3arf%15
IPv4 Address . . . . . : 192.168.2.1
Subnet Mask . . . . . : 255.255.255.252
Default Gateway . . . . . :

C:\Users\Administrator>ping 192.168.2.1

Pinging 192.168.2.1 with:32 bytes of data:
April 24, 2024 4:57:40 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:41 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:42 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:33 PM: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round-trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Administrator>ping 192.168.2.10

Pinging 192.168.2.10 with:32 bytes of data:
April 24, 2024 4:57:41 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:43 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:46 PM: bytes=32 time=1ms TTL=64
April 24, 2024 4:57:47 PM: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.2.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round-trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Administrator>

```

Figure 9: Successful ping from S-DC to 192.168.2.1 and 192.168.2.10

1.4 Significance of Baseline Validation

This stage was important because it established a valid operational baseline for the remainder of the assessment. Although a working network does not automatically mean that the environment is secure, later findings become far more meaningful once basic communication, services and routing have been shown to function correctly. In the SWANSEA.ACCA lab, the successful ping results demonstrated that the required hosts were reachable and that the network was operating as intended rather than failing due to simple configuration or connectivity problems. This means that the vulnerability analysis discussed later in the report can be interpreted as an assessment of the network's actual security posture, rather than as a consequence of basic operational faults. Such an approach is consistent with structured technical testing guidance, which emphasises that validation of the target environment should take place before deeper security conclusions are drawn [12].

Host	Validation Command	Purpose	Observed Outcome
S-DC	ipconfig	Confirm Windows host address	192.168.0.81 active
ClientS1	ipconfig	Confirm Windows client address	192.168.1.10 active

L-CD	ip a	Confirm Debian server address	10.86.10.193 active
Debian Client	ip a	Confirm GUI client address	192.168.0.86 active

Table 1.1: Task 1 validation summary

Task 2: Network Security Assessment and Vulnerability Analysis

2.1 Assessment Approach:

Following successful network validation, the SWANSEA.ACCA environment was examined from a cybersecurity perspective in order to identify the most significant risks within the network. The assessment concentrated on the assets and trust relationships that would have the greatest operational and security impact if compromised. These included the Windows Server 2022 domain controller, the DNS and DHCP services hosted on that server, the Debian systems connected within the wider environment, and the communication path linking the Swansea and London sites. This was an appropriate assessment strategy because the NIST Cybersecurity Framework 2.0 emphasises the need to identify critical assets, understand risk exposure and prioritise security decisions according to organisational importance.

The logic behind this approach is that the most critical systems should be assessed first, since compromise of those systems would have the widest effect across the environment. In the SWANSEA.ACCA lab, the domain controller represents the main trust anchor because it supports identity, authentication and core network services. At the same time, the Debian systems are also significant because they extend trust into a mixed operating system environment, while the site-to-site connection increases the effective network boundary and therefore the potential attack surface. For this reason, focusing the assessment on these high-value systems provides a more realistic and academically justified basis for vulnerability analysis and later hardening recommendations. [13]

2.1 Information Gathering with Nmap and db_nmap

Nmap and db_nmap were used from the Kali Linux system to carry out information gathering against the host identified as S-DC. The results confirmed that the target was active and exposed a range of services typically associated with a Windows domain controller, including DNS on port 53, Kerberos on port 88, LDAP on port 389, secure LDAP on port 636, SMB on port 445, and Global Catalog services on ports 3268 and 3269. These findings were important because they not only verified that the host was reachable, but also provided strong evidence that the machine was performing a central infrastructure role within the SWANSEA.ACCA environment. In this way, the scan supported both host identification and service enumeration, which are essential early steps in structured network security assessment [14].

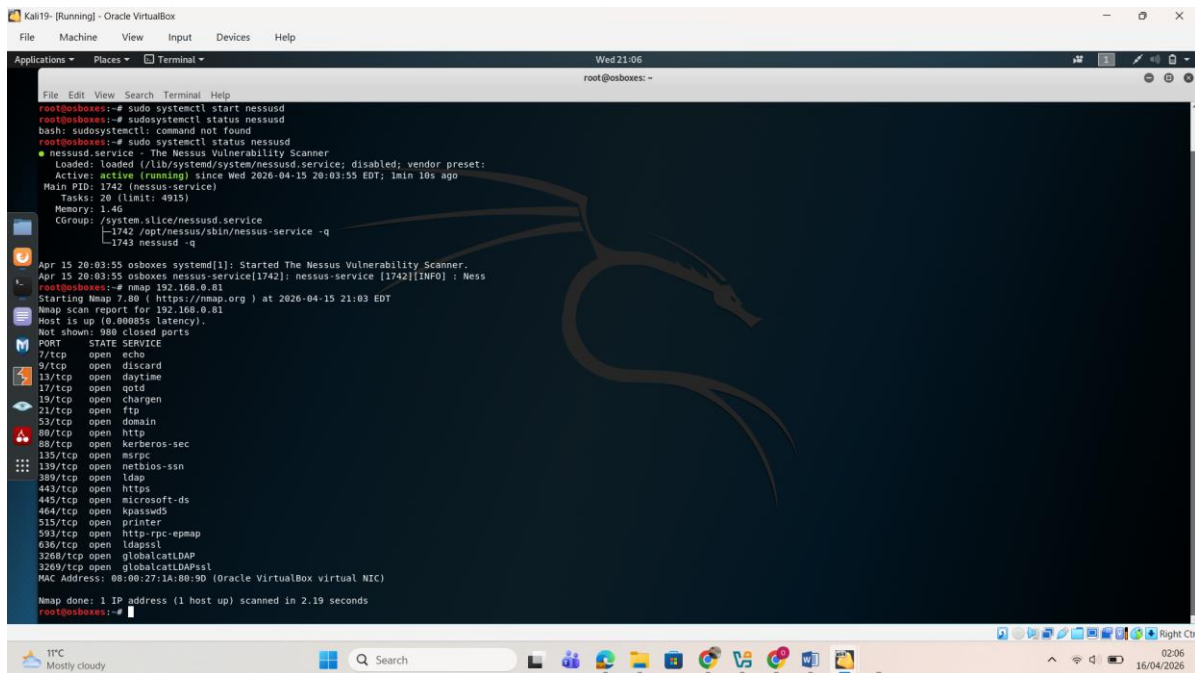


Figure 10: Nmap scan of S-DC showing open domain-controller-related services

db_nmap within Metasploit provided deeper service identification. It identified the target at 10.86.1.5 as Windows Server 2022 Datacenter and linked it to the swansea.acca domain. The same output also confirmed that SMB message signing was enabled and required, which is a positive control. At the same time, db_nmap showed anonymous FTP access and IIS support for the TRACE method, which informed the later risk assessment.

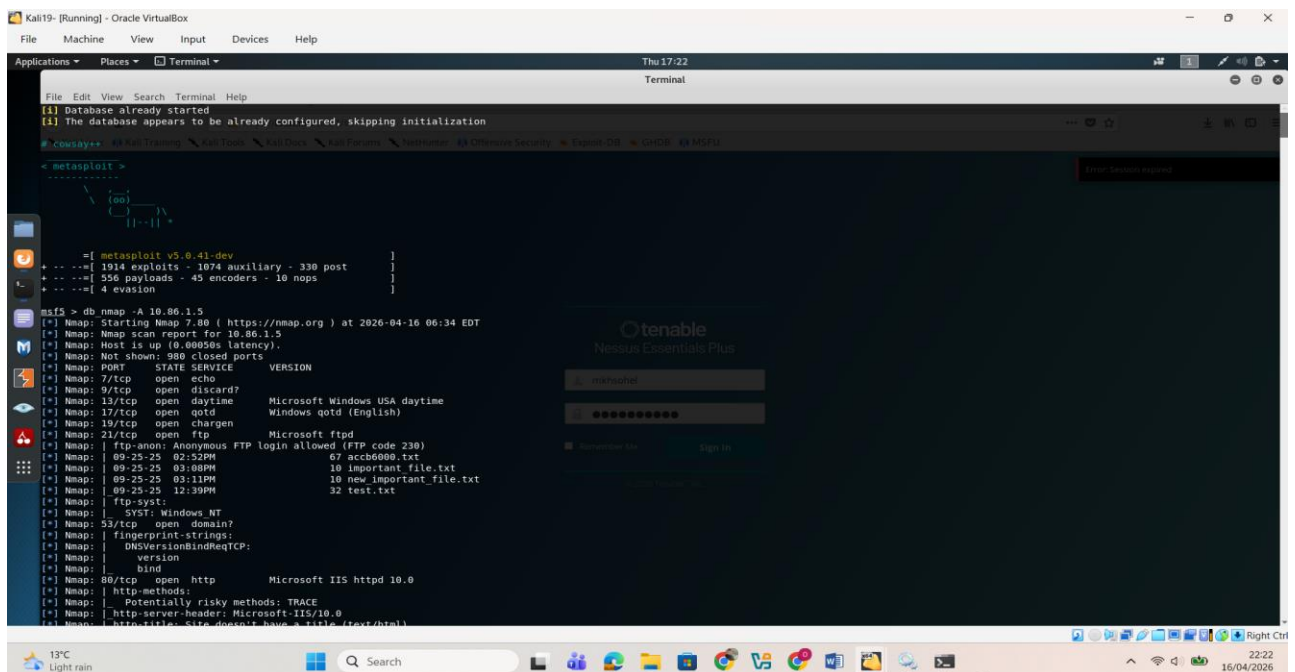


Figure 11: db_nmap output showing anonymous FTP access and IIS details on S-DC

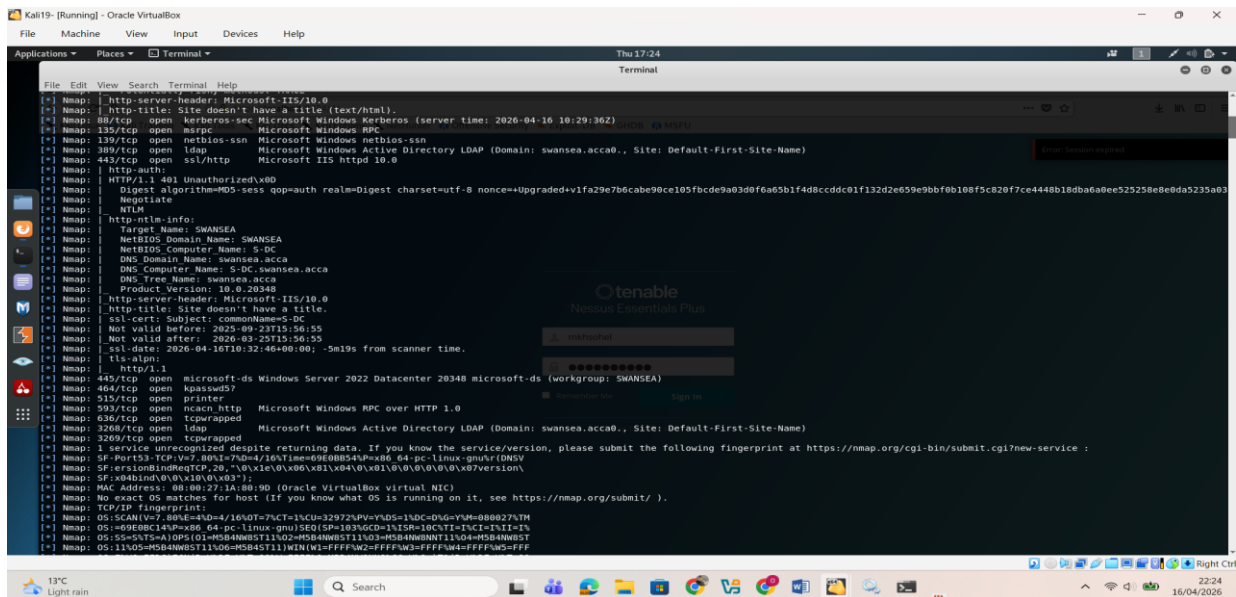


Figure 12: db_nmap advanced host discovery showing OS, host name and domain details

2.2 Vulnerability Assessment Using Nessus

Nessus Essentials was subsequently used to conduct an advanced vulnerability scan against the selected target host as part of the formal assessment stage. The scan summary confirmed that the process completed successfully for a single host, while the host-specific results page provided a detailed breakdown of the identified findings. Although authentication to the system was unsuccessful, meaning that the scan was carried out in a non-credentialed manner, the results still generated valuable evidence for the coursework. In particular, the scan was able to identify service-related weaknesses, protocol issues and other externally observable exposures, which were sufficient to support the vulnerability analysis and prioritisation presented later in this report [15].

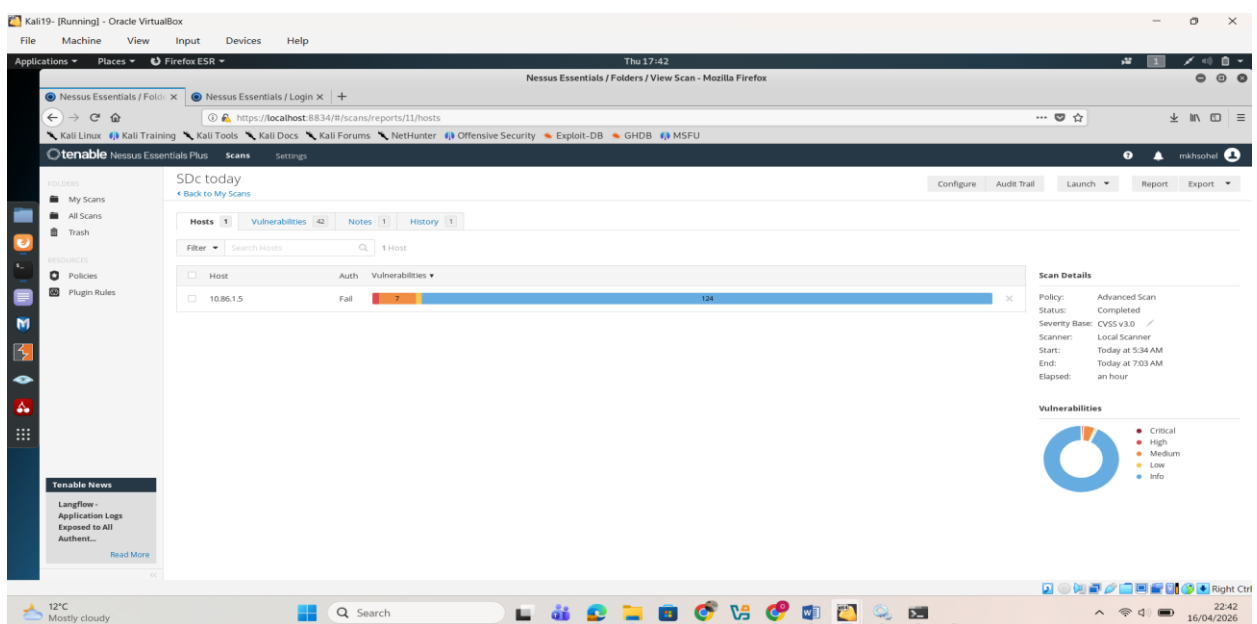


Figure 13: Nessus summary page for the completed S-DC scan

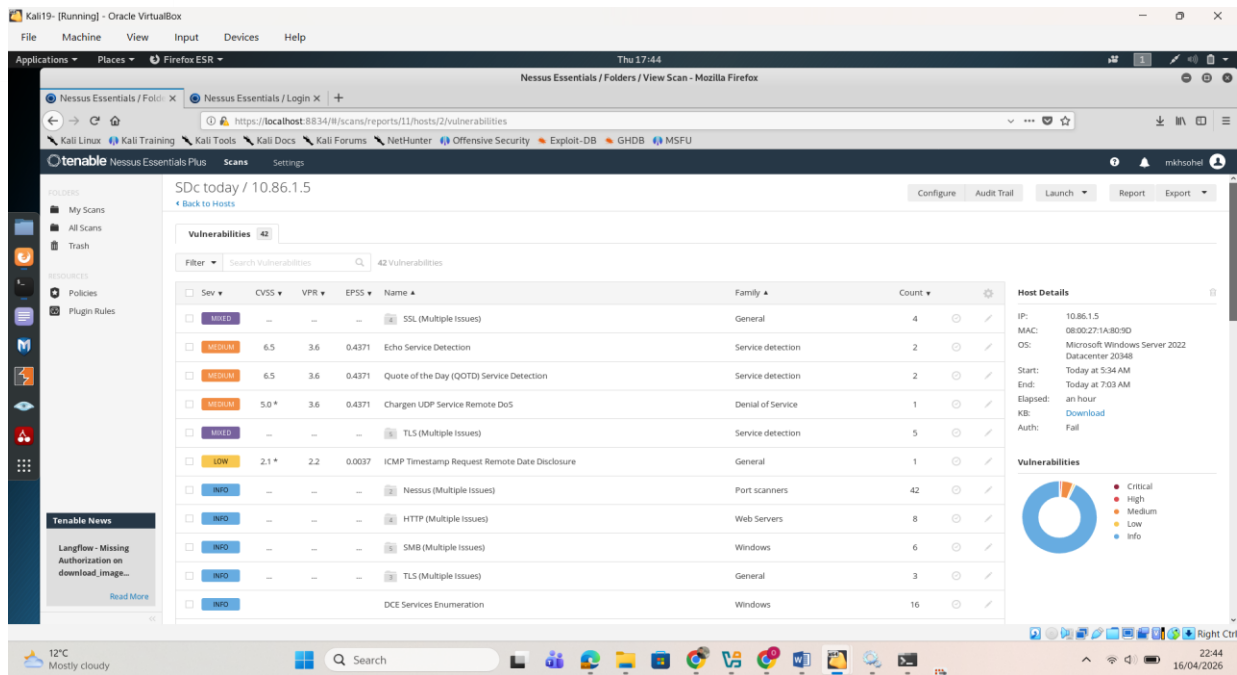


Figure 14: Nessus vulnerability list for S-DC / 10.86.1.5

The key findings shown by Nessus included SSL Medium Strength Cipher Suites Supported (SWEET32), TLS Version 1.0 Protocol Detection, TLS Version 1.1 Deprecated Protocol, Chargen UDP Service Remote DoS, Echo Service Detection, Quote of the Day Service Detection and ICMP Timestamp Request Remote Date Disclosure. The SSL page also clearly showed a high-severity issue for SWEET32, while the TLS page showed deprecated protocol support.

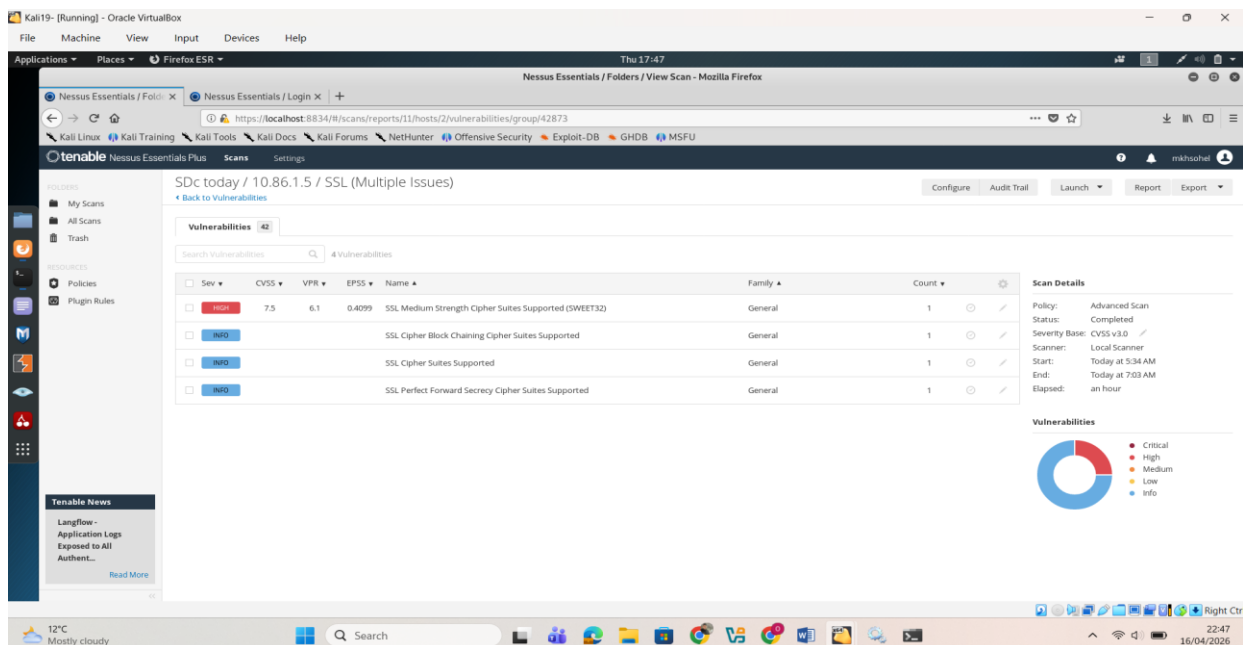


Figure 15: Nessus high-severity SSL finding showing SWEET32 cipher support

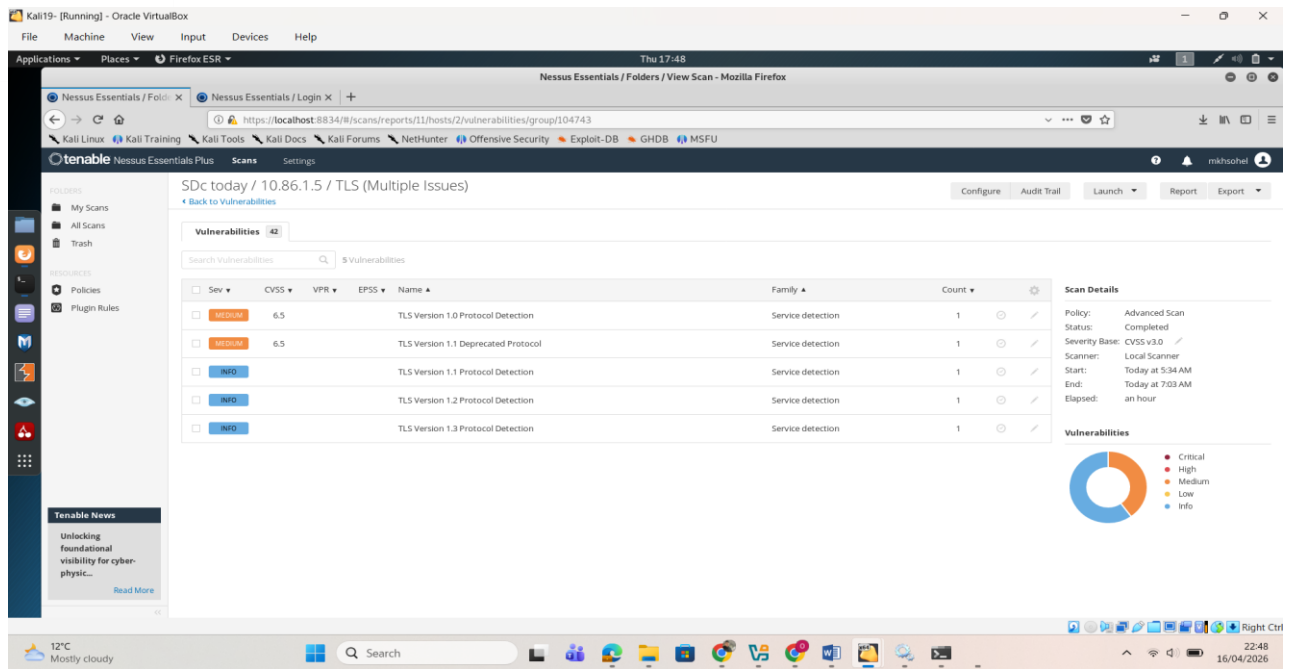


Figure 16: Nessus TLS findings showing deprecated protocol support

2.3 Final Vulnerability Set

The final vulnerability set presented below combines the severity ratings generated by the scanner with contextual judgement based on the role of the target as a central infrastructure server. This approach was used to ensure that the selected findings were not only technically identified, but also prioritised according to their likely operational and security impact within the SWANSEA.ACCA environment. As a result, the final set was structured to meet the coursework requirement of two high-risk, two medium-risk and two low-risk vulnerabilities in a way that reflects both scanner evidence and professional risk assessment practice [16].

ID	Vulnerability	Risk Level	Affected Area	Main Security Concern
V1	SSL Medium Strength Cipher Suites Supported (SWEET32)	High	S-DC TLS/SSL service	Weakens encrypted communication by allowing medium-strength ciphers
V2	Anonymous FTP access and risky HTTP TRACE method identified during	High	S-DC FTP and web services	Allows unauthenticated access and increases exposure through

	db_nmap enumeration			insecure legacy web behaviour
V3	TLS Version 1.0 Protocol Detection	Medium	TLS service on S-DC	Deprecated cryptographic protocol still supported
V4	Chargen UDP Service Remote DoS	Medium	Legacy network service on S-DC	Can be abused for denial-of-service or reflection-style traffic misuse
V5	ICMP Timestamp Request Remote Date Disclosure	Low	Network stack on S-DC	Reveals system time information useful for profiling
V6	Echo and QOTD legacy service exposure	Low	Legacy diagnostic services on S-DC	Unnecessary services increase attack surface and provide avoidable exposure

Table 2.1: Summary of identified vulnerabilities and recommended controls

2.4 Discussion of the Six Vulnerabilities

The two high-risk vulnerabilities identified in this assessment were closely linked to the security of the central server. The first was the support for medium-strength SSL cipher suites, specifically the SWEET32 issue identified by Nessus. This was considered high risk because the S-DC server performs a critical infrastructure role within the SWANSEA.ACCA environment, meaning that weaknesses in transport security could affect sensitive communications more seriously than they would on a standard endpoint. The second high-risk issue came from the db_nmap results, which indicated anonymous FTP access together with support for the HTTP TRACE method. On a server responsible for core services, the presence of such insecure or legacy behaviour increases unnecessary exposure and creates additional opportunities for misuse.

The medium-risk findings were deprecated TLS protocol support and the exposure of the Chargen UDP service. Deprecated TLS support is significant because older protocol versions no longer provide the level of cryptographic protection expected in modern secure environments and should therefore be replaced by stronger, up-to-date alternatives. Similarly, Chargen is a legacy network service with little practical value in a secure enterprise setting and may be abused in denial-of-service or reflection-based misuse. Although these weaknesses are

less severe than the high-risk issues, they still represent important security concerns because they weaken the resilience of the server.

The low-risk findings were ICMP timestamp disclosure and the presence of Echo and QOTD services. Individually, these weaknesses are unlikely to cause immediate compromise, but they still matter from a security perspective because they reveal unnecessary information and expose outdated functionality that does not appear essential to the system's intended role. Taken together, these low-risk findings suggest that the server presents more information and more legacy service exposure than is strictly required, which reinforces the need for stronger service reduction and hardening.

Task 3: Ethical Hacking Highlights

3.1 Ethical Scope and Metasploit Use

Metasploit was used in a controlled and ethically authorised manner as part of the SWANSEA.ACCA lab assessment. Its role in this exercise was not to support random or destructive exploitation, but to assist with reconnaissance, db_nmap-based information gathering and structured review of relevant modules. This approach was appropriate because the purpose of the lab was to validate identified weaknesses, understand their likely impact and support later remediation, rather than to carry out uncontrolled attacks. Using Metasploit in this limited and evidence-based way therefore aligned with the principles of responsible security testing and professional ethical hacking practice [17].

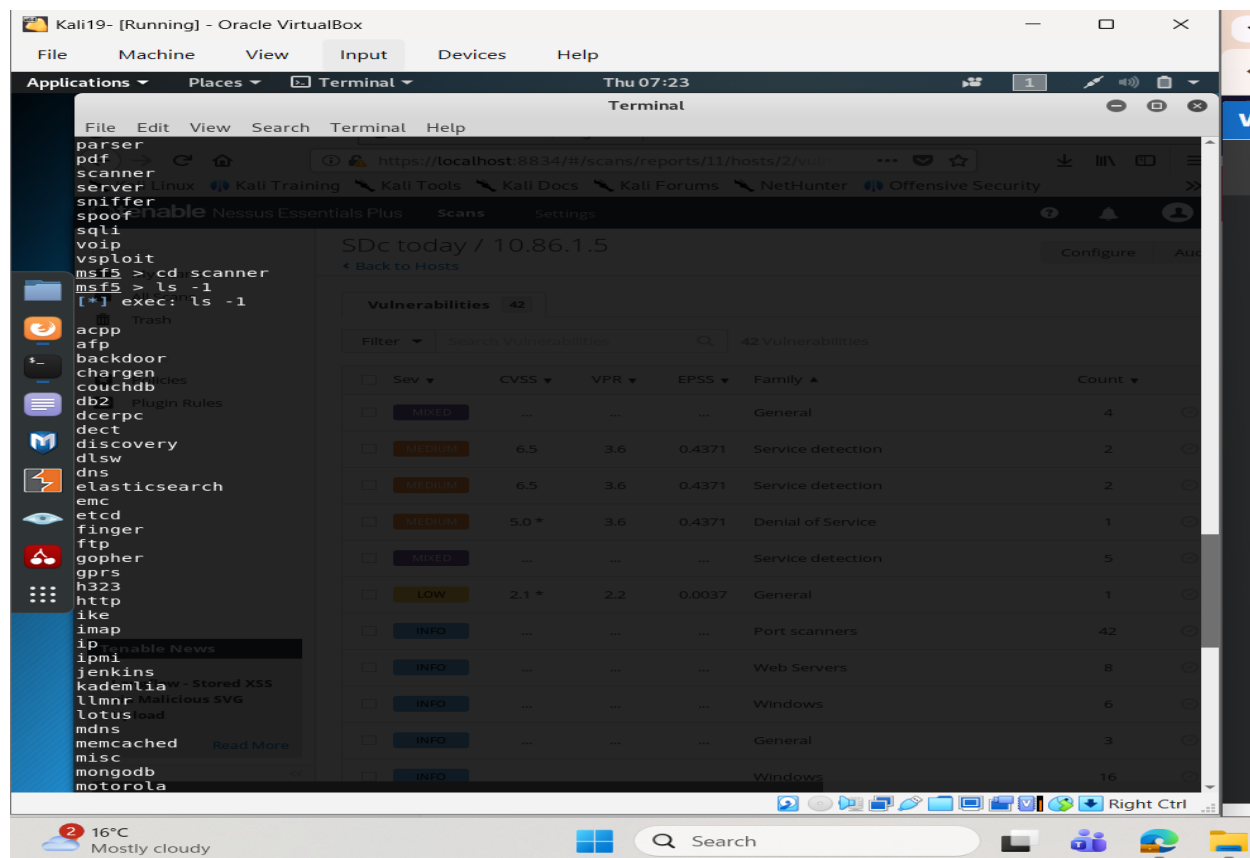


Figure 17: Listing Metasploit auxiliary scanner modules

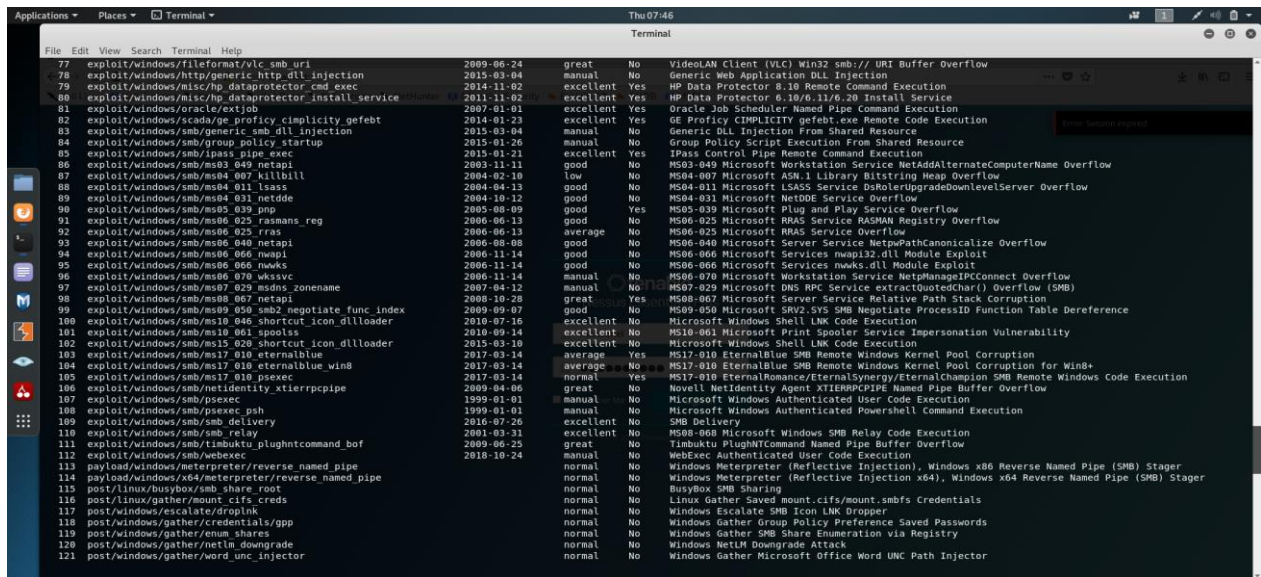


Figure 18: SMB-related Metasploit module search results

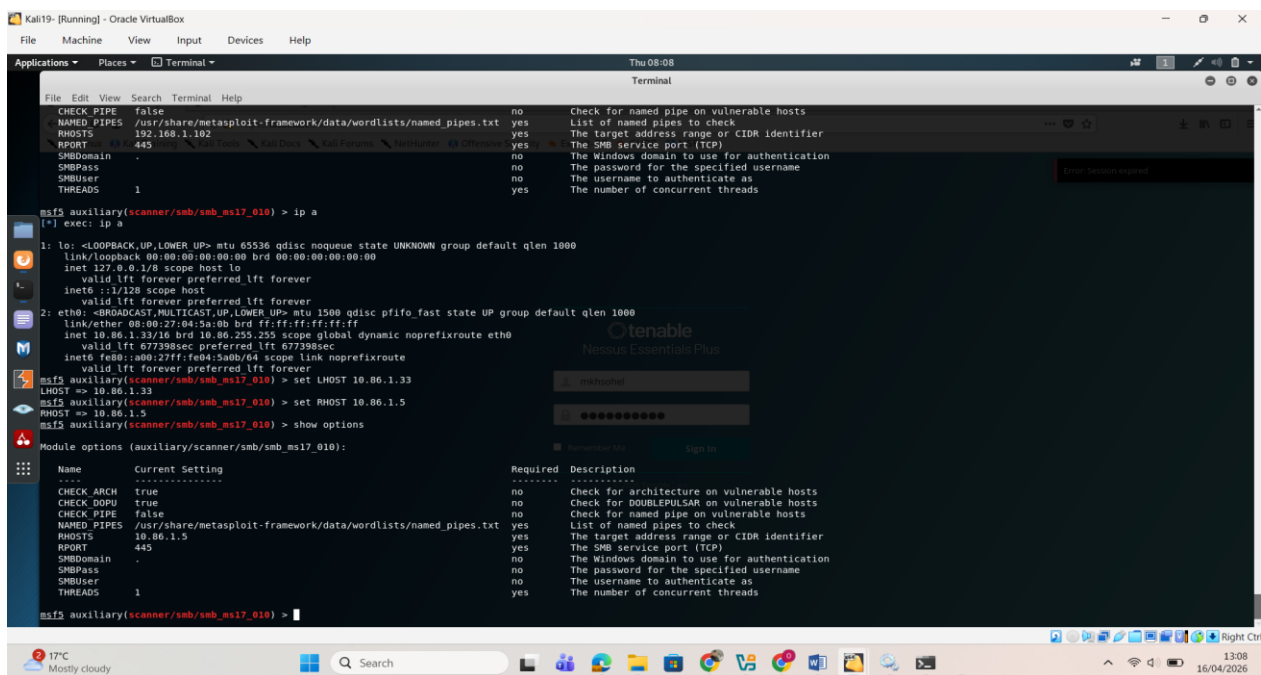


Figure 19: Metasploit auxiliary module options showing LHOST and RHOST configuration

3.2 Exploitation and Post-Exploitation Highlights

The most appropriate vulnerabilities to highlight in the demonstration were the high-severity TLS weakness, one medium-severity legacy service issue, and one low-severity information disclosure issue. In the presentation, these findings should be discussed at a high level rather than as a harmful step-by-step exploitation guide. The key academic purpose is to explain why insecure cipher support, deprecated protocols and unnecessary exposed services are important

from a cybersecurity perspective, particularly because unresolved weaknesses of this kind can affect the confidentiality, integrity and availability of systems and services [18].

Task 4: Preventive Security Measures

4.1 Windows Server Hardening:

The primary hardening priority in the SWANSEA.ACCA environment is the Windows Server 2022 domain controller, as it performs the most critical infrastructure roles within the network. Since this server hosts Active Directory Domain Services, DNS and DHCP, it functions as the central point of trust, authentication and network management for both sites. For this reason, its security configuration must be treated as a priority rather than as a routine administrative task. In practical terms, hardening the server should include enforcing stronger password and account policies, restricting administrative privileges, reviewing and tightening firewall rules, disabling unnecessary services, and applying secure baseline settings appropriate to a domain controller. Strengthening these areas reduces the likelihood that weaknesses on a single host will undermine the wider security of the SWANSEA.ACCA environment.[19]

4.2 Protocol Protection:

Protocol protection is the second priority. SMB signing should be enforced to protect message integrity, while LDAP signing and channel binding should also be enabled to strengthen directory-related communication. These measures are valuable because they protect the trust relationships that normal Windows administration depends on. Rather than assuming that internal communications are safe by default, the network should require important traffic to meet stronger integrity standards [20]

4.3 Debian Hardening:

The Debian systems should be hardened to the same standard as the Windows hosts. The Securing Debian Manual explains that Debian security includes hardening the default installation, reducing unnecessary exposure and maintaining a secure operating environment. For this lab, that means updating packages, reducing unneeded services, reviewing access controls and ensuring that Linux domain integration is configured carefully. Because the Debian systems are part of the wider trusted environment, weak Linux security would still affect the overall network posture [21].

4.4 Patch and Vulnerability Management:

Patch management should be treated as a continuous security process rather than an occasional maintenance task. In the SWANSEA.ACCA environment, this is important because outdated systems may continue operating normally while still exposing known weaknesses. NIST guidance explains that effective patch management should be handled as preventive maintenance and should include verification after updates have been deployed. Alongside this, vulnerability scanning is valuable for identifying weak configurations, missing patches and exposed services that may otherwise go

unnoticed. For this reason, tools such as Nessus and Greenbone OpenVAS are appropriate in this assignment because they provide structured vulnerability identification and reporting to support security review and remediation planning [22].

4.5 Hardening and Remediation Strategy

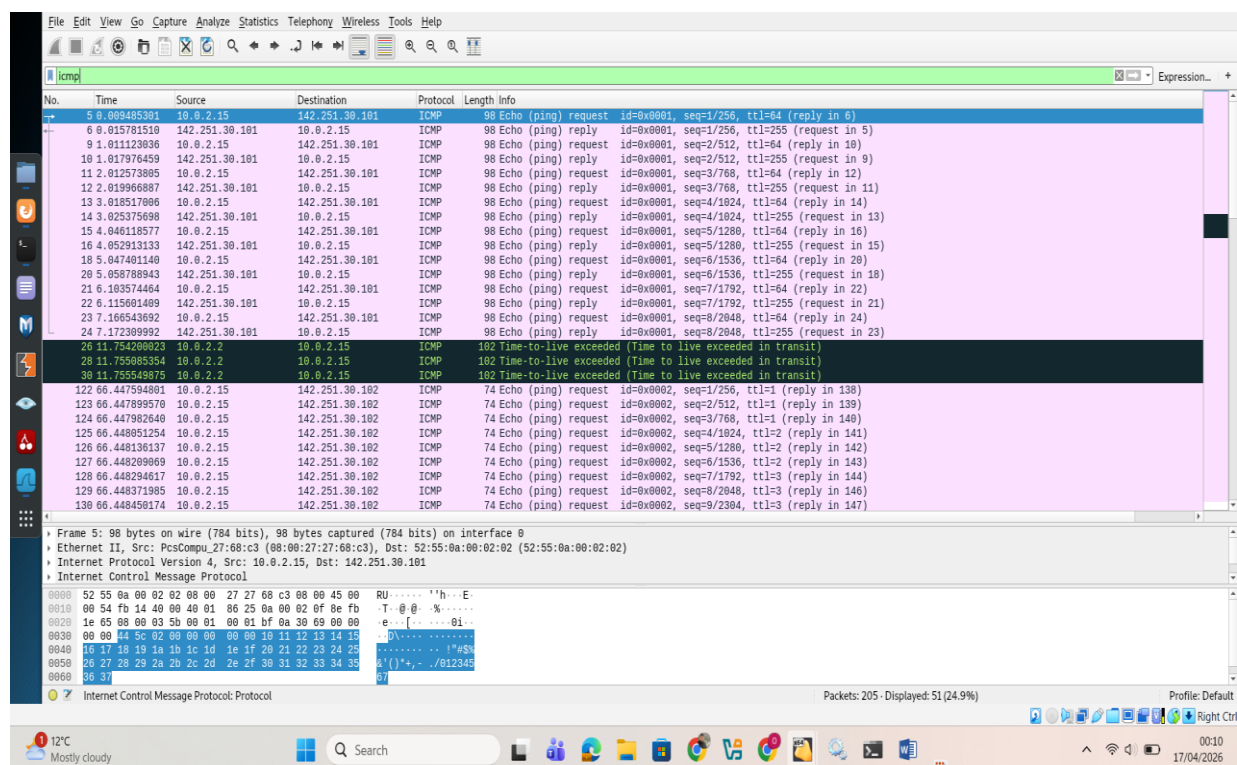
The recommended hardening strategy for the SWANSEA.ACCA environment focused on reducing unnecessary exposure, strengthening transport security and improving overall visibility. The highest priority is to disable medium-strength cipher suites and remove support for legacy protocols such as TLS 1.0 and TLS 1.1 wherever possible, as these weaken the protection of network communications. The next step is to restrict or disable anonymous FTP access, remove risky HTTP methods such as TRACE, and turn off unnecessary legacy services including Echo, QOTD and Chargen. Additional measures should include Linux-side hardening, continuous patch management and repeated rescanning after changes have been applied, so that any reduction in attack surface can be verified and documented as part of the ongoing security improvement process [23].

Security Measure	Related Findings	Purpose
Windows TLS/SSL hardening	V1, V3	Remove weak ciphers and deprecated TLS versions
Service reduction on S-DC	V2, V4, V6	Disable unnecessary FTP, Echo, QOTD and Chargen exposure
Web server hardening	V2	Disable risky HTTP methods and review IIS configuration
Linux hardening	Operational support	Reduce Linux-side exposure and strengthen administration
Patch management and rescanning	V1-V6	Confirm remediation and reduce known vulnerabilities

Table 4.1: Preventive security measures

4.6 Logging, Monitoring and Traffic Visibility

Wireshark was used to observe and analyse live network activity within the lab environment. The filtered capture showed ICMP echo requests and replies between hosts, demonstrating that packet-level monitoring can be used to verify communication paths, confirm connectivity and investigate unusual behaviour. This is important because network visibility complements host hardening by allowing administrators to inspect traffic in real time rather than relying only on endpoint configuration. Wireshark's documentation describes it as a network packet analyzer that presents captured packet data in as much detail as possible, while its display filter functionality allows specific protocols such as ICMP to be isolated for clearer analysis. As a result, the Wireshark evidence in this assignment supports the view that effective monitoring improves the ability to detect, understand and respond to abnormal or suspicious traffic within the SWANSEA.ACCA environment [24].



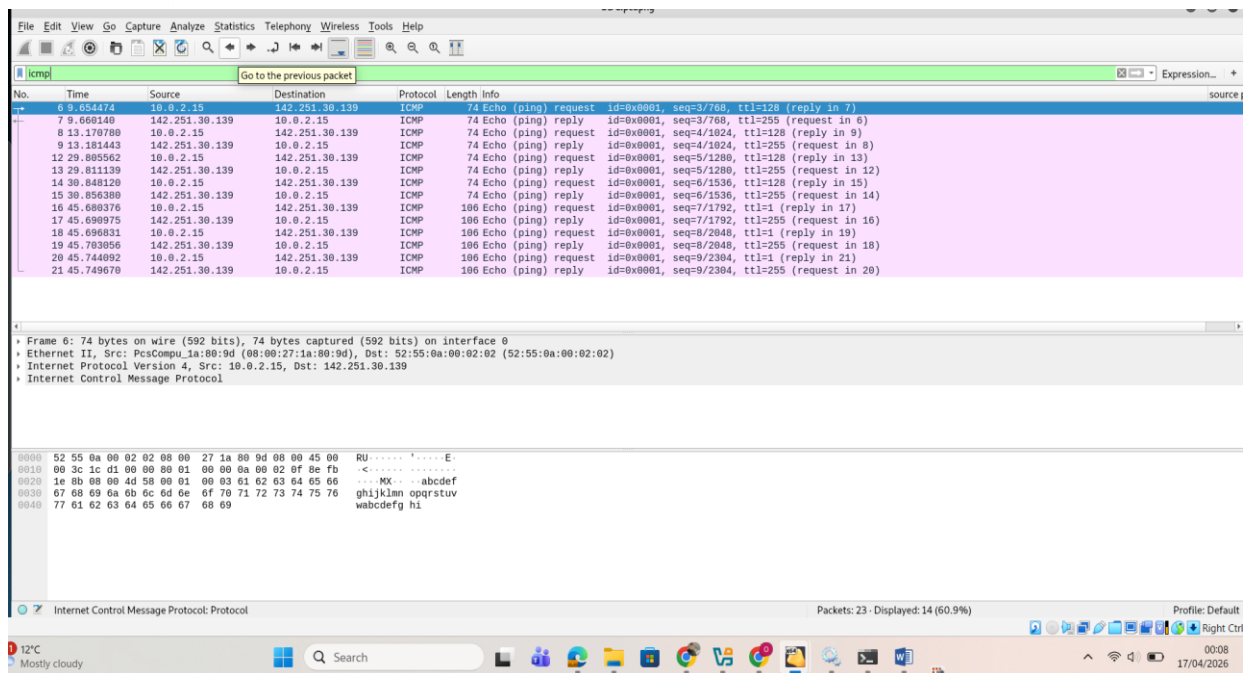


Figure 20: Wireshark capture showing monitored ICMP traffic between lab hosts

Conclusion

The SWANSEA.ACCA environment was shown to be operational through interface checks and successful ping testing, and it was then assessed using Nmap, db_nmap, Nessus, Metasploit module review and Wireshark monitoring. The captured evidence demonstrated that the main target behaved like a Windows Server 2022 domain controller within the swansea.acca environment. The final vulnerability set combined scanner results with contextual judgement to produce two high-risk, two medium-risk and two low-risk findings suitable for the portfolio.

Overall, the report shows that an operational network is not necessarily a secure one. The environment functioned correctly, but it still exposed weak transport security, unnecessary legacy services and unnecessary information disclosure. By tightening TLS, reducing exposed services, hardening web and FTP configuration, continuing patch management and improving visibility, the security posture of the SWANSEA.ACCA environment can be improved significantly.

References

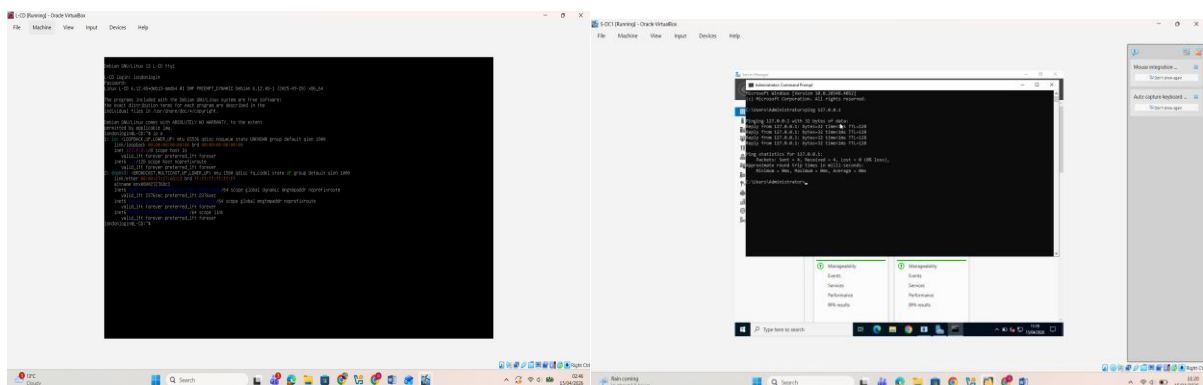
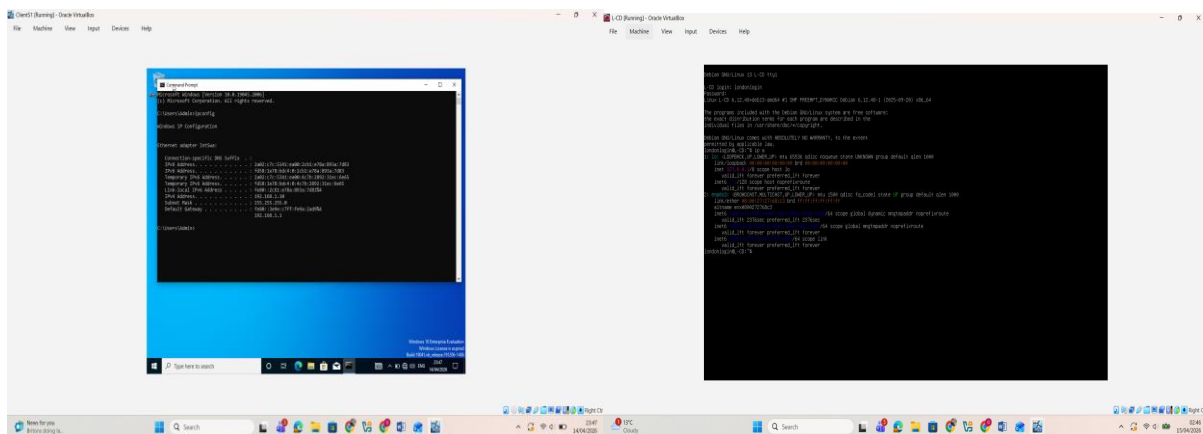
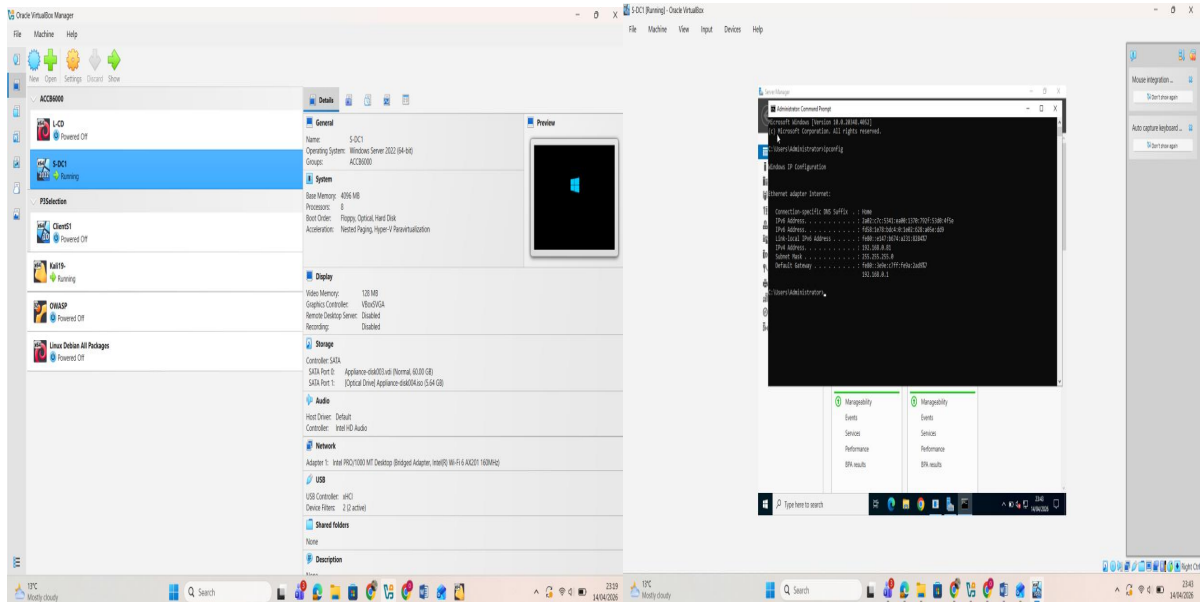
- [1] National Institute of Standards and Technology, *The NIST Cybersecurity Framework (CSF) 2.0*, NIST CSWP 29, Gaithersburg, MD, USA, Feb. 2024. [Accessed: Mar. 18, 2026].
- [2] M. Souppaya and K. Scarfone, *Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology*, NIST Special Publication 800-40 Rev. 4, Gaithersburg, MD, USA, Apr. 2022. [Accessed: Mar. 18, 2026].
- [3] Microsoft, “Active Directory Domain Services overview,” *Microsoft Learn*, Mar. 11, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview>. [Accessed: Apr. 10, 2026].
- [4] J. Fernández-Sanguino Peña, *Securing Debian Manual*, Debian Project, ver. 3.19. [Online]. Available: <https://www.debian.org/doc/manuals/securing-debian-manual/index.en.html>. [Accessed: Apr. 12, 2026].
- [5] OpenVPN, “Site-to-Site Networking,” *OpenVPN*. [Online]. Available: <https://openvpn.net/use-cases/site-to-site-vpn/>. [Accessed: Apr. 24, 2026].
- [6] K. Scarfone, M. Souppaya, A. Cody, and A. Orebaugh, *Technical Guide to Information Security Testing and Assessment*, NIST SP 800-115, Gaithersburg, MD, USA, Sep. 2008. [Accessed: Apr. 13, 2026].
- [7] National Institute of Standards and Technology, *The NIST Cybersecurity Framework (CSF) 2.0*, NIST CSWP 29, Gaithersburg, MD, USA, Feb. 2024. [Accessed: Apr. 12, 2026].
- [8] Microsoft, “Active Directory Domain Services overview,” *Microsoft Learn*, Mar. 11, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview>. [Accessed: Apr. 14, 2026].
- [9] Microsoft, “Domain Name System (DNS) - Windows Server,” *Microsoft Learn*, May 15, 2025, and Microsoft, “What is DHCP Server in Windows Server?,” *Microsoft Learn*, May 13, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/networking/technologies/dhcp/dhcp-top>. [Accessed: Apr. 14, 2026].
- [10] OpenVPN, “Site-to-Site Networking,” *OpenVPN*. [Online]. Available: <https://openvpn.net/use-cases/site-to-site-vpn/>. [Accessed: Apr. 07, 2026].
- [11] K. Scarfone, M. Souppaya, A. Cody and A. Orebaugh, “Technical Guide to Information Security Testing and Assessment,” NIST SP 800-115, Sep. 2008. [Accessed: Apr. 07, 2026].
- [12] K. Scarfone, M. Souppaya, A. Cody and A. Orebaugh, “Technical Guide to Information Security Testing and Assessment,” NIST SP 800-115, Sep. 2008. [Accessed: Apr. 08, 2026].

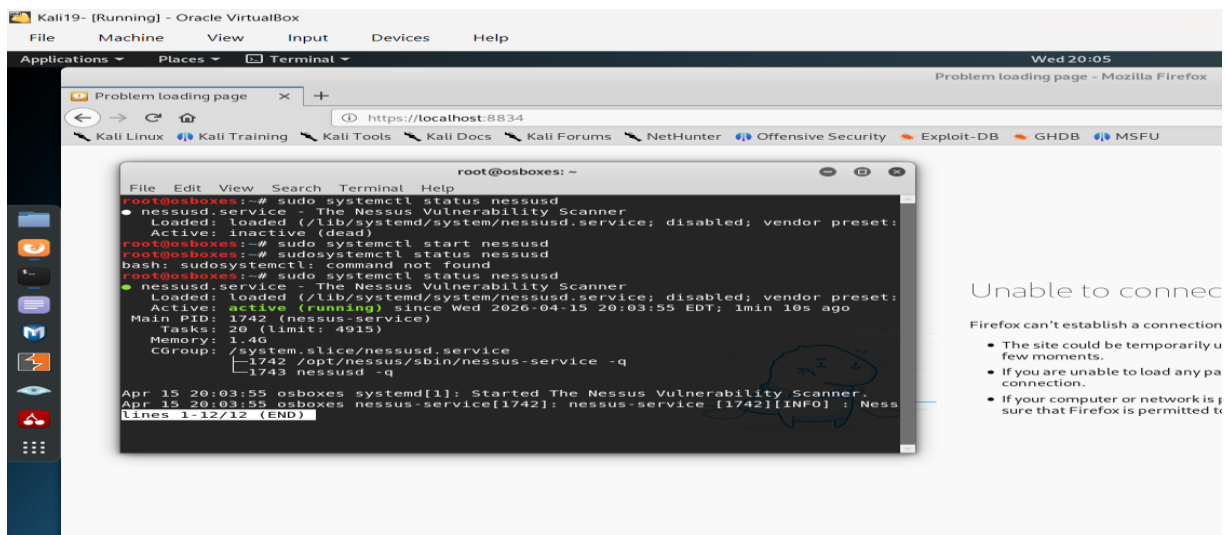
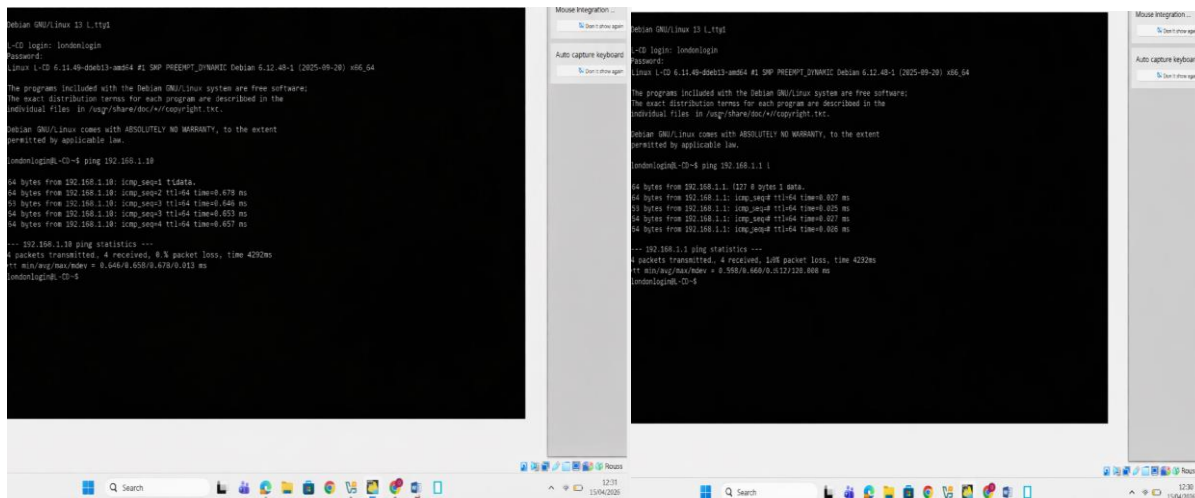
- [13] National Institute of Standards and Technology, “The NIST Cybersecurity Framework (CSF) 2.0,” NIST CSWP 29, Feb. 2024. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf>. [Accessed: Apr. 09, 2026].
- [14] Nmap Project, “Nmap Reference Guide,” *Nmap Network Scanning*. [Online]. Available: <https://nmap.org/book/man.html>. [Accessed: Apr. 09, 2026].
- [15] Tenable, “Nessus Vulnerability Scanner,” *Tenable*. [Online]. Available: <https://www.tenable.com/products/nessus>. [Accessed: Apr. 07, 2026].
- [16] C. McNab, *Network Security Assessment: Know Your Network*, 3rd ed. Sebastopol, CA, USA: O’Reilly Media, 2016. [Accessed: Apr. 10, 2026].
- [17] D. Kennedy, J. O’Gorman, D. Kearns, and M. Aharoni, *Metasploit: The Penetration Tester’s Guide*. San Francisco, CA, USA: No Starch Press, 2011.
- [18] J. P. A. Yaacoub, O. Salman, H. N. Noura, and A. Chehab, “Ethical hacking for IoT: Security issues, challenges, and open problems,” *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 1–21, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2667345223000238>. [Accessed: Apr. 11, 2026].
- [19] J. Mallery, *Hardening Windows Systems*. Boca Raton, FL, USA: CRC Press, 2016. [Online]. Available: *insert URL here*. [Accessed: Apr. 12, 2026].
- [20] J. R. Vacca, *Computer and Information Security Handbook*, 3rd ed. Amsterdam, The Netherlands: Elsevier, 2017.
- [21] Debian Project, *Securing Debian Manual*. [Online]. Available: <https://www.debian.org/doc/manuals/securing-debian-manual/index.en.html>. [Accessed: Apr. 11, 2026].
- [22] M. Souppaya and K. Scarfone, “Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology,” NIST SP 800-40 Rev. 4, Apr. 2022. [Online]. Available: <https://csrc.nist.gov/pubs/sp/800/40/r4/final>. [Accessed: Apr. 12, 2026].
- [23] K. Scarfone, *Guide to General Server Security*, NIST SP 800-123, Gaithersburg, MD, USA, Jul. 2008. [Online]. Available: <https://csrc.nist.gov/pubs/sp/800/123/final>. [Accessed: Apr. 15, 2026].
- [24] Wireshark Foundation, “What is Wireshark?,” in *Wireshark User’s Guide*. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html. [Accessed: Apr. 15, 2026].

Appendix A - Authenticity and evidence note

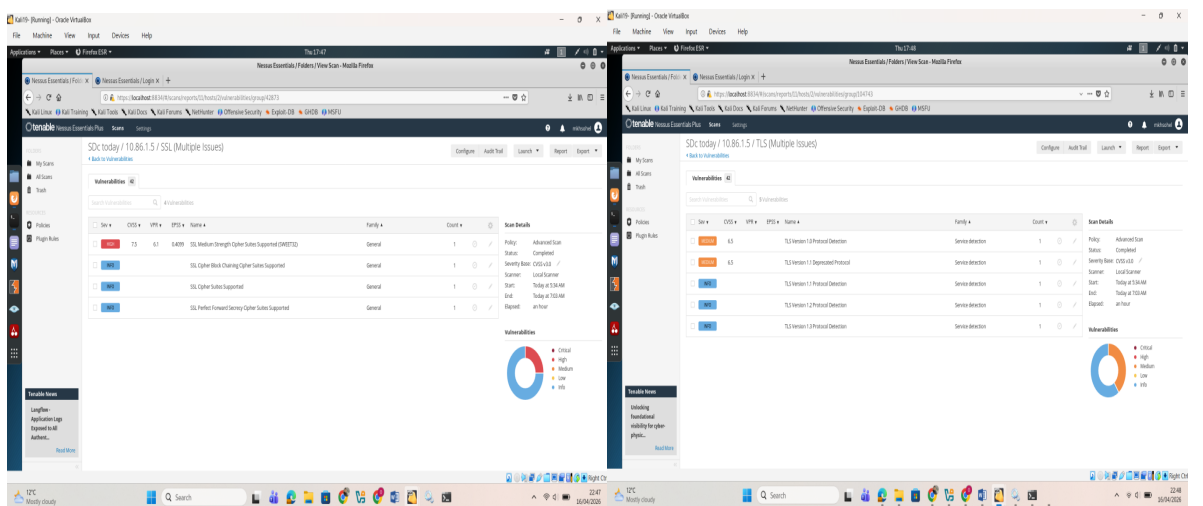
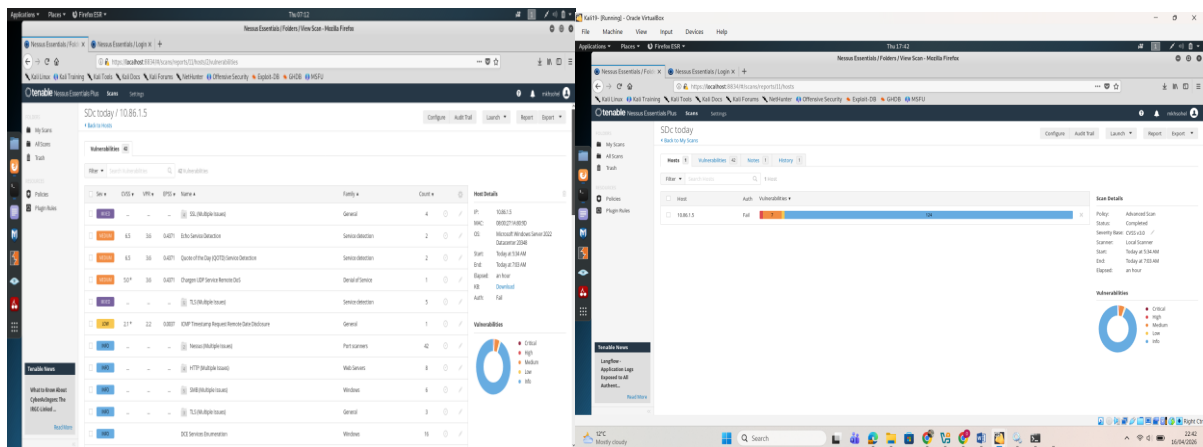
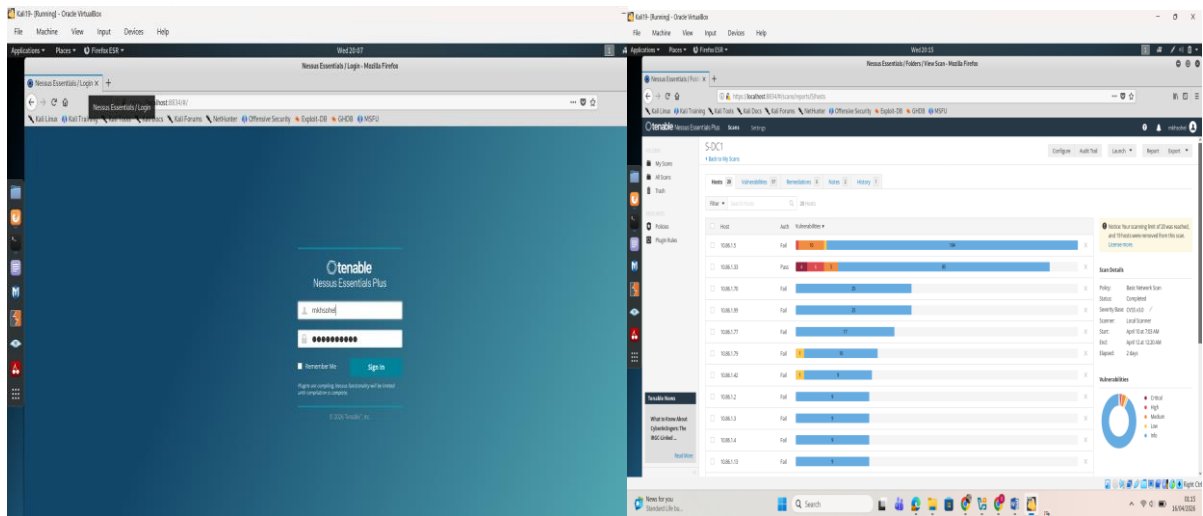
provided live screenshots and exported evidence from the lab environment. The inserted images and scan outputs should therefore be treated as the primary evidence pack for submission and presentation.

Network Setup and Ping:





Appendix B: Network Security Assessment:



S-DC1 / 10.86.133
[Back to Hosts](#)

Configure Audit Trail Launch Report Export

Vulnerabilities 68

Filter Search Vulnerabilities 68 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count	
MIXED	...	...	...	4 Apache Log4j (Multiple Issues)	Misc.	7	
HIGH	7.8	9.7	0.9231	Linux Sudo Privilege Escalation (Out-of-bounds Write)	Misc.	1	
HIGH	7.5	3.6	0.0004	Python Library Brotli <= 1.1.0 DoS	Misc.	1	
MIXED	...	...	...	3 Imagemagick (Multiple Issues)	Misc.	3	
MIXED	...	...	...	2 Nodejs Node.js (Multiple Issues)	Misc.	2	
MEDIUM	5.9	4.4	0.0001	nginx 1.3.0 < 1.28.2 / 1.29.x < 1.29.5 SSL Upstream Injection	Web Servers	2	
MIXED	...	...	...	4 SSL (Multiple Issues)	General	4	
MIXED	...	...	...	2 Nginx (Multiple Issues)	Web Servers	2	
INFO	...	...	...	6 SSH (Multiple Issues)	General	6	
INFO	...	...	...	2 Apache HTTP Server (Multiple Issues)	Web Servers	2	
INFO	...	...	...	2 HTTP (Multiple Issues)	Web Servers	2	

Host: 10.86.133

Host Details

IP: 10.86.133
MAC: 08:00:27:04:5A:0B
OS: Linux Kernel 5.2.0-kali2-amd64
Start: April 10 at 7:03 AM
End: April 10 at 7:11 AM
Elapsed: 8 minutes
KB: [Download](#)
Auth: Pass

Vulnerabilities



Right Ctrl

01:34
18/04/2026

S-DC1 / 10.86.133
[Back to Hosts](#)

Configure Audit Trail Launch Report Export

Vulnerabilities 68

Filter Search Vulnerabilities 68 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count	
MIXED	...	...	...	4 Apache Log4j (Multiple Issues)	Misc.	7	
HIGH	7.8	9.7	0.9231	Linux Sudo Privilege Escalation (Out-of-bounds Write)	Misc.	1	
HIGH	7.5	3.6	0.0004	Python Library Brotli <= 1.1.0 DoS	Misc.	1	
MIXED	...	...	...	3 Imagemagick (Multiple Issues)	Misc.	3	
MIXED	...	...	...	2 Nodejs Node.js (Multiple Issues)	Misc.	2	
MEDIUM	5.9	4.4	0.0001	nginx 1.3.0 < 1.28.2 / 1.29.x < 1.29.5 SSL Upstream Injection	Web Servers	2	
MIXED	...	...	...	4 SSL (Multiple Issues)	General	4	
MIXED	...	...	...	2 Nginx (Multiple Issues)	Web Servers	2	
INFO	...	...	...	6 SSH (Multiple Issues)	General	6	
INFO	...	...	...	2 Apache HTTP Server (Multiple Issues)	Web Servers	2	
INFO	...	...	...	2 HTTP (Multiple Issues)	Web Servers	2	

Host: 10.86.133

Host Details

IP: 10.86.133
MAC: 08:00:27:04:5A:0B
OS: Linux Kernel 5.2.0-kali2-amd64
Start: April 10 at 7:03 AM
End: April 10 at 7:11 AM
Elapsed: 8 minutes
KB: [Download](#)
Auth: Pass

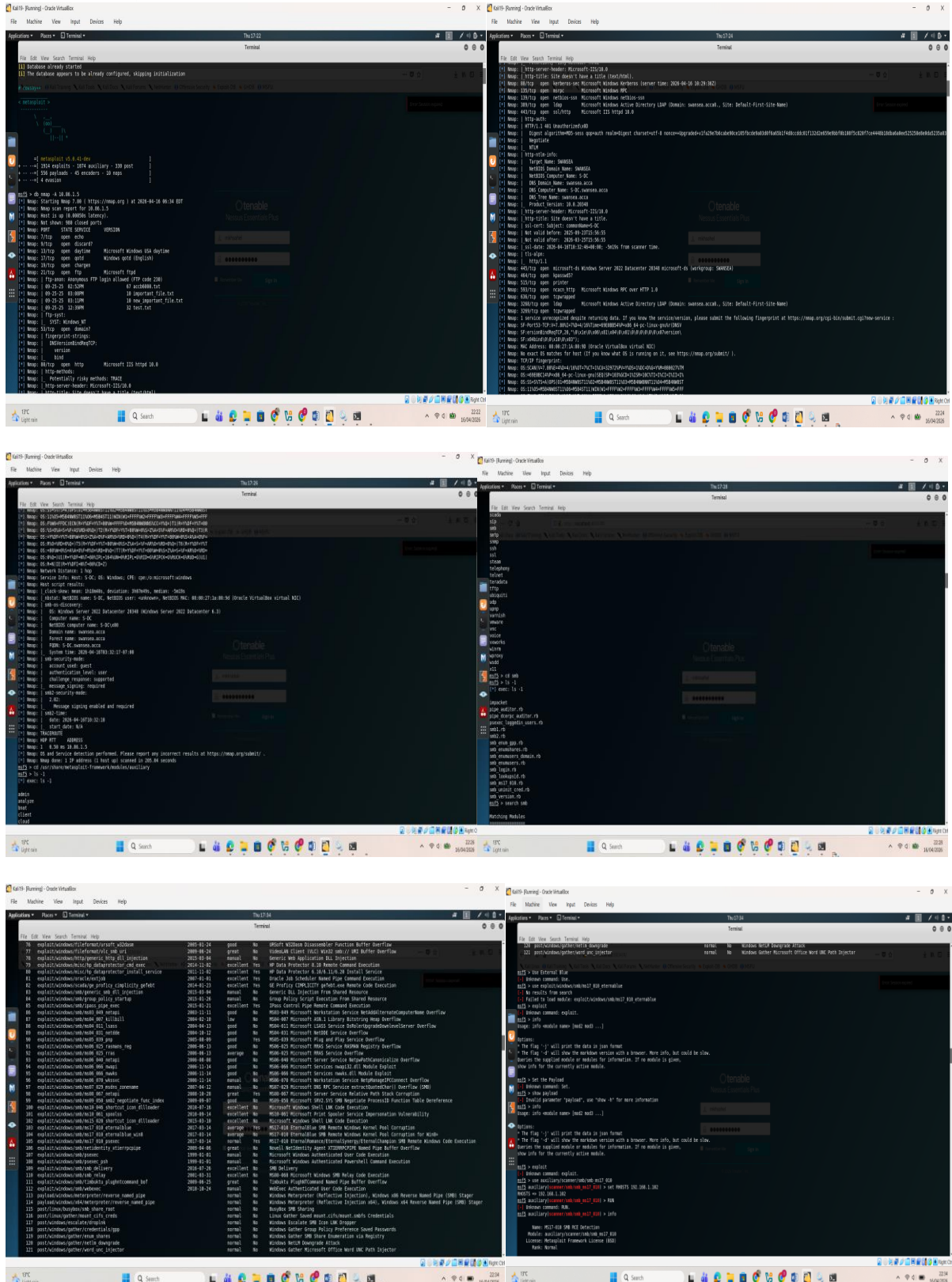
Vulnerabilities

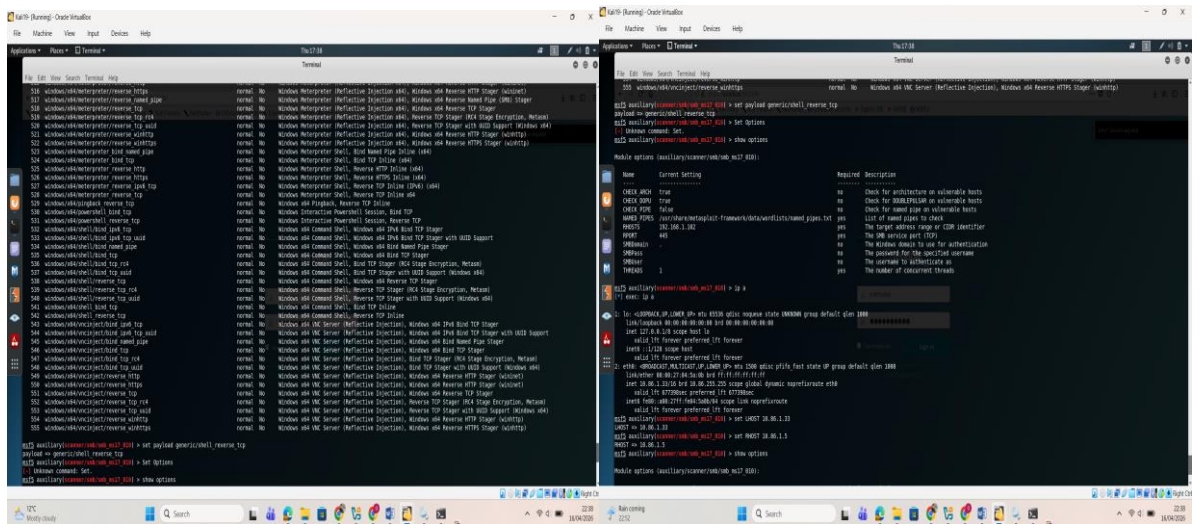
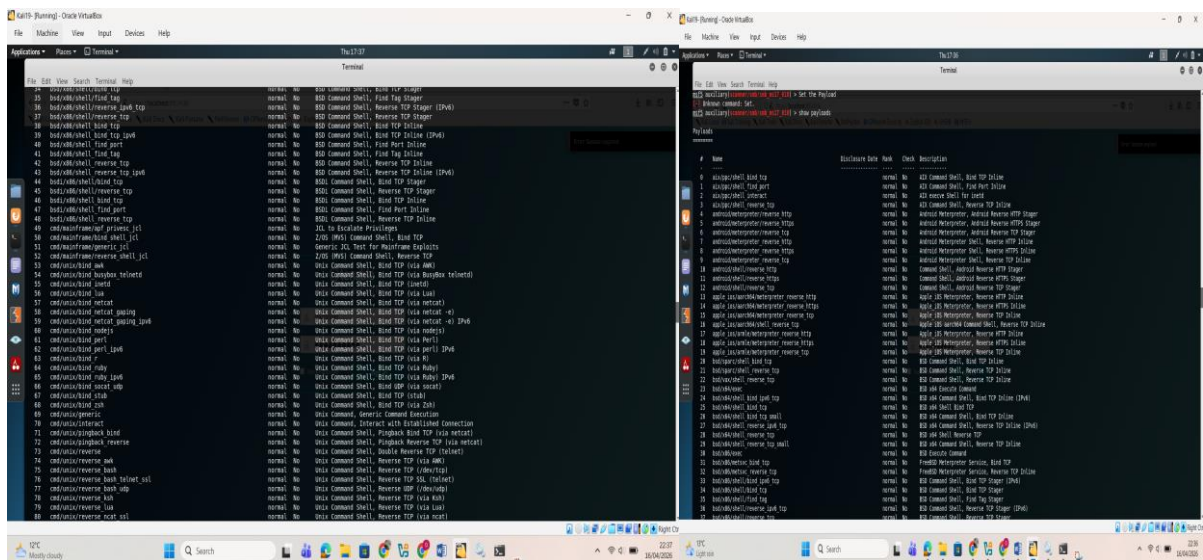
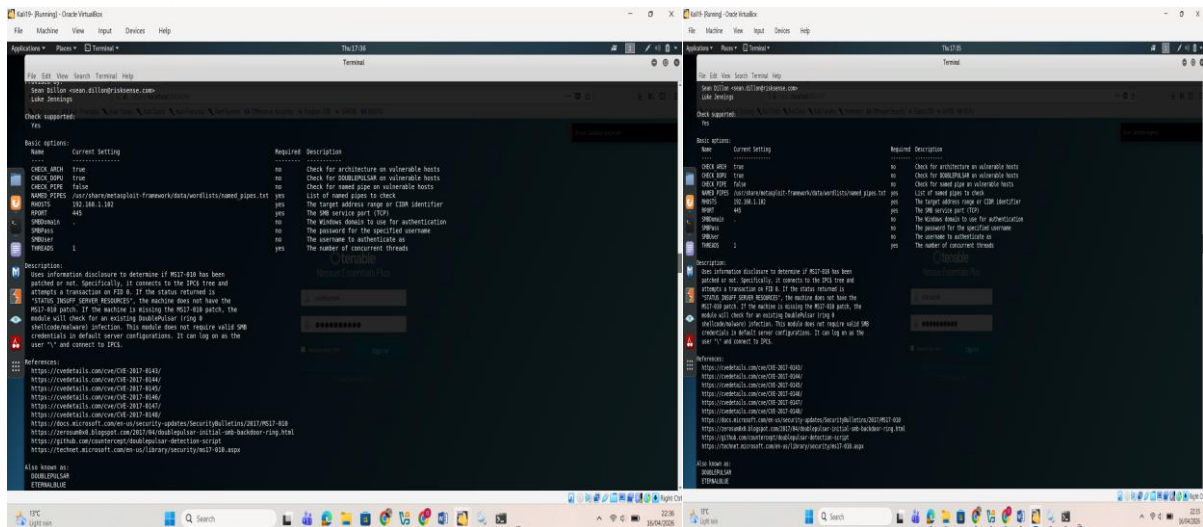


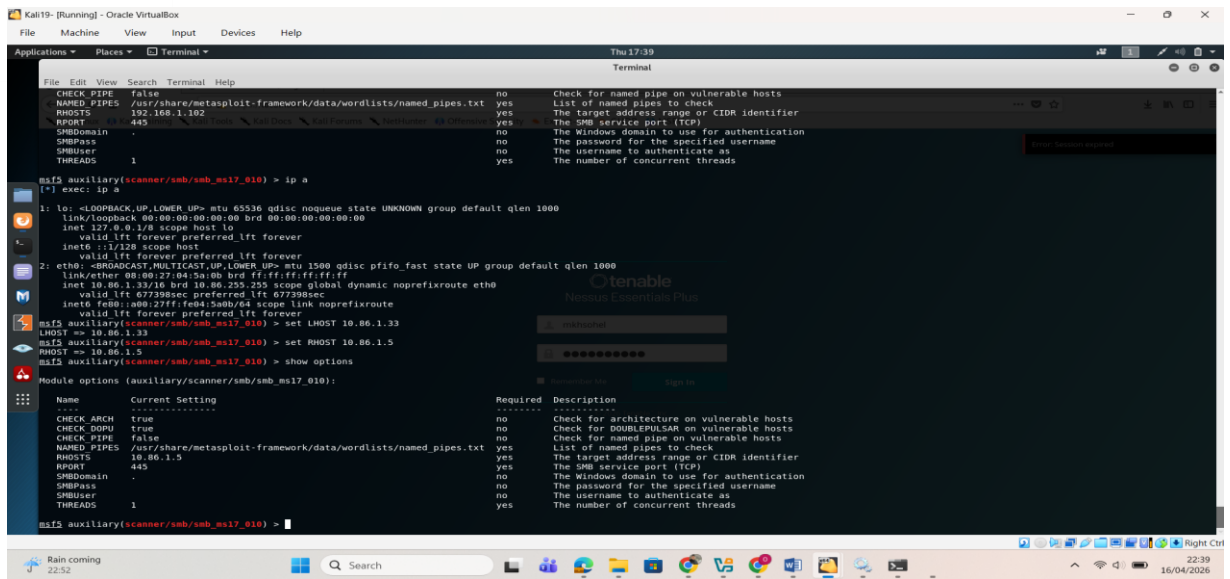
Right Ctrl

01:34
18/04/2026

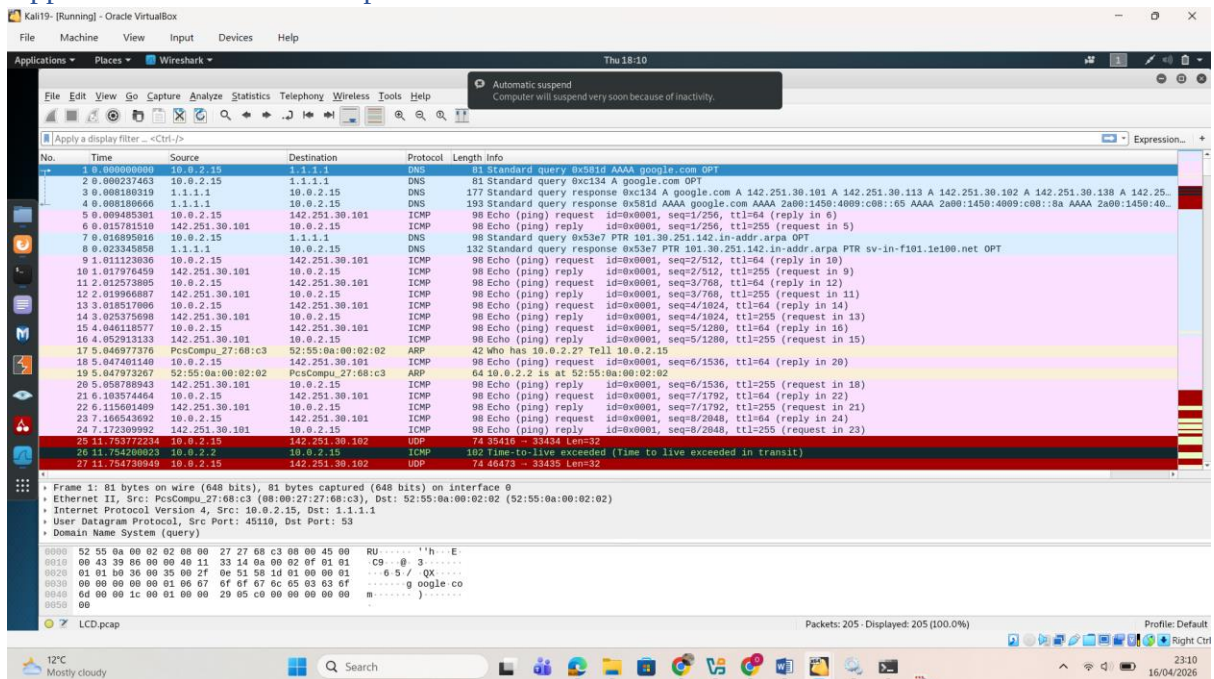
Appendix C: Dbnmap an metaexploit:







Appendix D: WireShark Capture



Kali19: [Running] - Oracle VirtualBox

File Machine View Input Devices Help

Applications Places Wireshark Thu 18:14

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

LCD.pcap

icmp

No.	Time	Source	Destination	Protocol	Length	Info
0	0.000000	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=1/256, ttl=64 (reply in 0)
1	0.015781510	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=1/256, ttl=255 (request in 5)
2	0.011123830	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=2/512, ttl=64 (reply in 10)
3	0.01976459	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=2/512, ttl=255 (request in 9)
4	0.012573805	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=3/768, ttl=64 (reply in 12)
5	0.019666887	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=3/768, ttl=255 (request in 11)
6	0.018517806	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=4/1824, ttl=64 (reply in 14)
7	0.025373690	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=4/1824, ttl=255 (request in 13)
8	0.046118577	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=5/1280, ttl=64 (reply in 16)
9	0.052913133	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=5/1280, ttl=255 (request in 15)
10	0.047481140	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=6/1536, ttl=64 (reply in 20)
11	0.050780943	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=6/1536, ttl=255 (request in 18)
12	0.103574464	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=7/1792, ttl=64 (reply in 22)
13	0.115601409	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=7/1792, ttl=255 (request in 21)
14	0.160543602	10.0.2.15	142.251.30.101	ICMP	98	Echo (ping) request id=0x0001, seq=8/2048, ttl=64 (reply in 24)
15	0.172099992	142.251.30.101	10.0.2.15	ICMP	98	Echo (ping) reply id=0x0001, seq=8/2048, ttl=255 (request in 23)
16	0.1754200023	10.0.2.15	10.0.2.15	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
17	0.1756003354	10.0.2.15	10.0.2.15	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
18	0.1755549875	10.0.2.15	10.0.2.15	ICMP	102	Time-to-live exceeded (Time to live exceeded in transit)
19	0.447594801	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=1/256, ttl=1 (reply in 138)
20	0.447899570	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=2/512, ttl=1 (reply in 139)
21	0.447982640	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=3/768, ttl=1 (reply in 140)
22	0.448051254	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=4/1824, ttl=2 (reply in 141)
23	0.448136137	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=5/1280, ttl=2 (reply in 142)
24	0.448209069	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=6/1536, ttl=2 (reply in 143)
25	0.448294617	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=7/1792, ttl=3 (reply in 144)
26	0.448371985	10.0.2.15	142.251.30.102	ICMP	74	Echo (ping) request id=0x0002, seq=8/2048, ttl=3 (reply in 146)

Frame 5: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
Ethernet II, Src: PcsCompu.27:68:c3 (08:00:27:27:68:c3), Dst: 52:55:0a:00:02:02 (52:55:0a:00:02:02)
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 142.251.30.101
Internet Control Message Protocol

0000 52 55 0a 00 02 02 00 00 27 27 68 c3 08 00 45 00 RU.....'h...E:
0010 00 54 fb 14 40 00 40 01 86 25 0a 00 02 0f 8e fb .T.@.%.....
0020 1e 65 88 00 63 5b 00 01 00 01 bf 0a 39 69 00 00 .e[...]...01..
0030 00 00 44 5c 02 00 00 00 00 00 19 11 12 13 14 15 .D\.....
0040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25!*\$%&
0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 &'()*+,-./012345

Packets: 205 - Displayed: 51 (24.9%)

Profile: Default

12°C Mostly cloudy

Search

23:14 16/04/2026

The End.