Windows Workstation Purple Team Attack and Detection Lab

A hands-on Purple Team lab simulating Windows reconnaissance, attack, privilege escalation, detection, and remediation in an isolated environment.

Prepared By:

Charlotte Galloway

Assessment Date:

September 7, 2026

Environment:

Isolated Virtual Lab

Tools:

Kali Linux · Nmap · SMBClient · FreeRDP · Wireshark · Windows Event Viewer · Windows Defender · VirtualBox

Table of Contents

1.0 EXECUTIVE SUMMARY

This project developed a controlled Windows workstation environment to demonstrate how offensive security techniques can be identified and investigated through defensive monitoring. The goal was to simulate a realistic endpoint attack while safely observing the activity generated by that attack.

The lab consisted of an attacker system and a Windows workstation connected through an isolated virtual network. The target was first examined through reconnaissance and service enumeration to establish a baseline of the system. A deliberately vulnerable service configuration was then introduced to create an attack path for privilege escalation.

Following the attack, Windows security logs and system telemetry were analyzed to identify evidence of the activity. This included examining events associated with service modification, process execution, and privilege-related behavior. The observed telemetry was correlated with the actions performed during the attack to demonstrate how defenders can detect and investigate suspicious endpoint activity.

The results demonstrate the core principles of purple-team security by connecting offensive techniques with defensive detection. The project shows how a vulnerable service configuration can create a privilege-escalation opportunity while also producing observable artifacts that defenders can use during an investigation.

Overall, the lab demonstrates the importance of combining secure system configuration with effective monitoring and detection. By understanding both how an attack is performed and what evidence it leaves behind, security teams can improve their ability to identify, investigate, and respond to endpoint threats.

2.0 LAB ENVIORNMENT

2.1 ARCHITECTURE

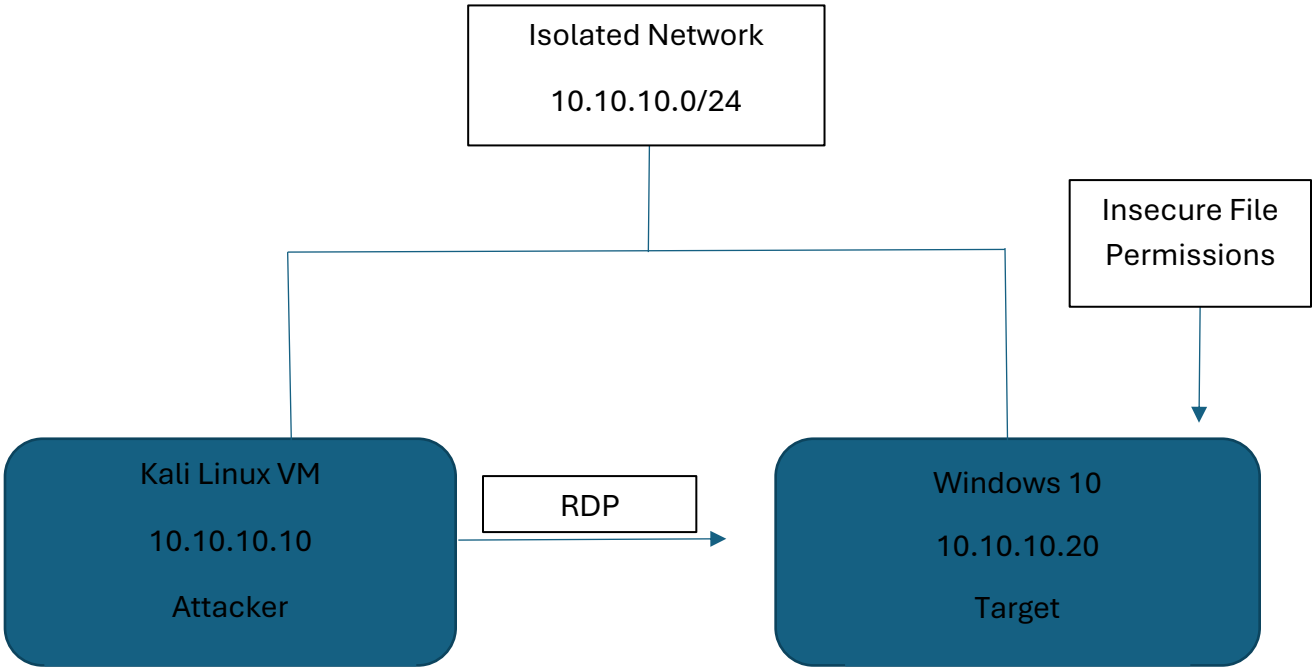

Figure 1. Windows Workstation Purple Team Architecture

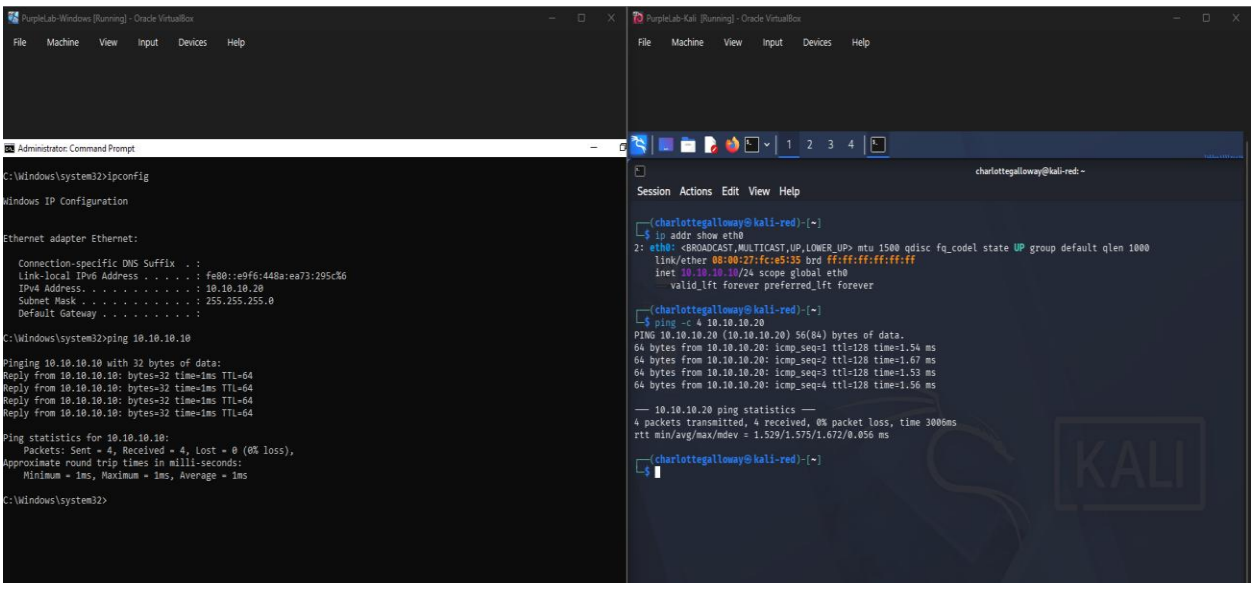

Figure 2. Isolated Network Connectivity between Kali Linux Attacker VM and Windows 10 target VM

2.2 TOOLS

Several tools were used throughout the laboratory to perform reconnaissance, attack simulation, system configuration, and defensive analysis.

- **VirtualBox** — Used to create and manage the isolated virtual machines and laboratory network.
- **Windows 11** — Served as the target workstation for the attack and detection exercises.
- **Kali Linux** — Served as the attacker machine for reconnaissance and security testing.
- **Nmap** — Used for network discovery, port scanning, and service enumeration.
- **Windows Command Prompt / PowerShell** — Used to configure the Windows workstation, manage services, and execute attack-related commands.
- **Windows Event Viewer** — Used to review Windows security and system events generated during testing.
- **Wireshark** — Used when applicable to observe and analyze network traffic within the laboratory environment.

Together, these tools provided the capabilities necessary to conduct the attack, observe its effects, and analyze the resulting security telemetry.

3.0 METHODOLOGY

3.1 RECONNAISSANCE

Reconnaissance began by verifying connectivity between the attacker and target systems using ping. Once communication was confirmed, Nmap was used to enumerate the target workstation at 10.10.10.20. The scan confirmed that the host was active and identified exposed services, including TCP port 631, associated

with CUPS, and TCP port 53, associated with the system's DNS resolver. These results established the target's initial attack surface and provided a baseline for the subsequent vulnerability assessment and attack phase.

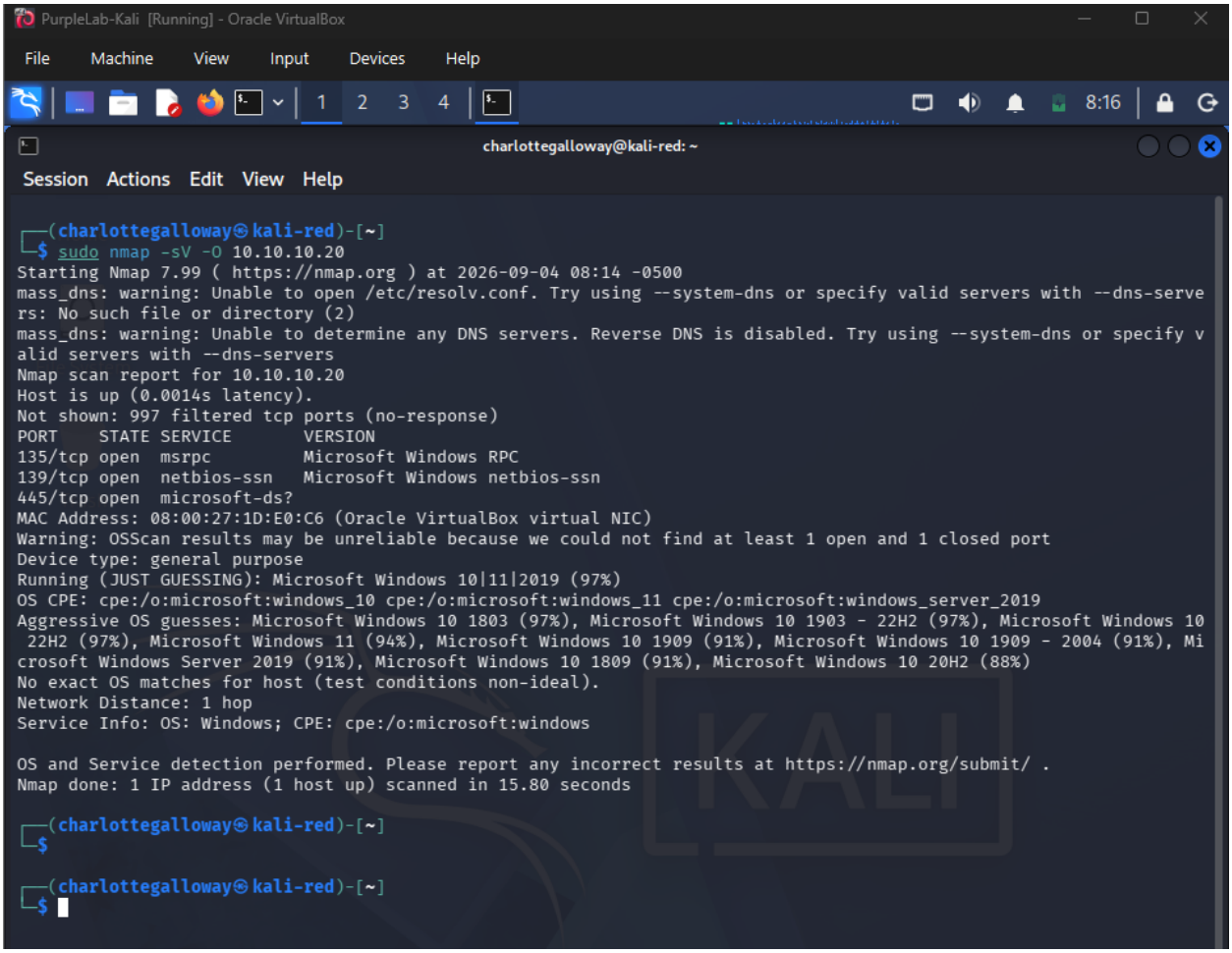

Figure 3. Initial Reconnaissance and service enumeration of Windows target.

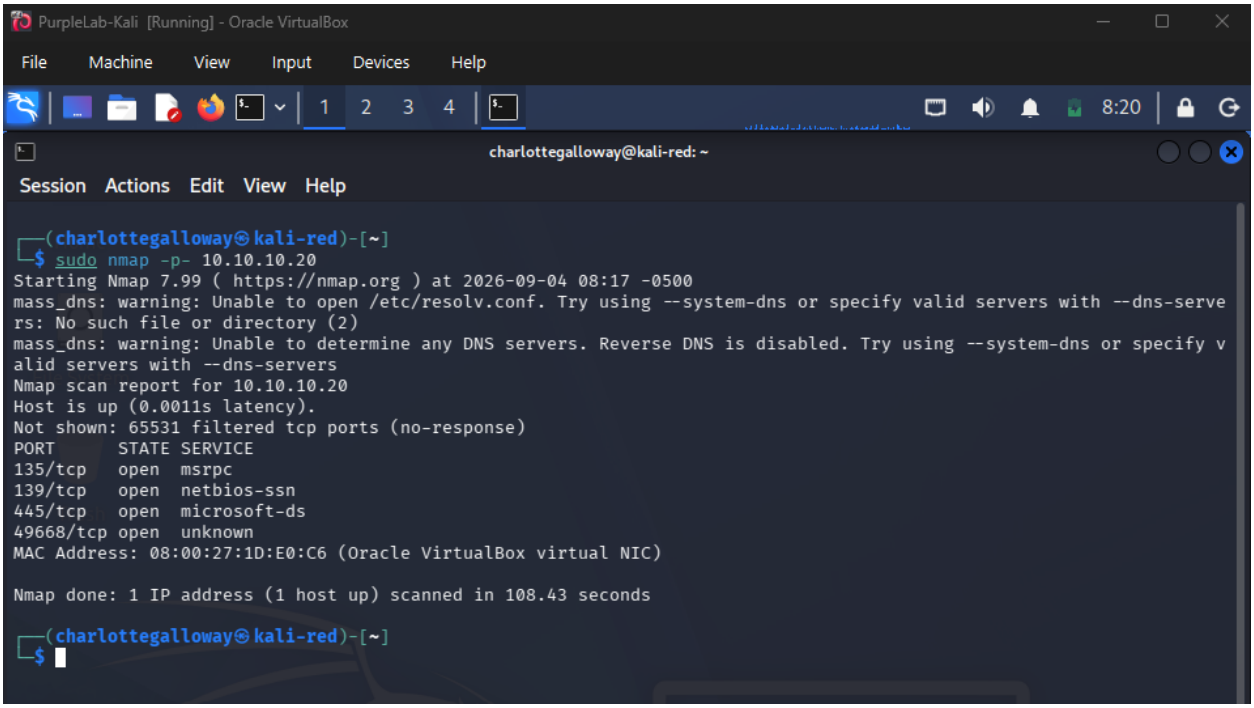

Figure 4. Full TCP port scan identifying exposed services on the windows target

3.2 ATTACK & PRIVLEGE ESCALATION

Following reconnaissance, the identified attack path was investigated by examining the target's service and associated file permissions. An insecure file permission configuration was identified that allowed the vulnerable service's executable or related files to be modified by an unauthorized user. This misconfiguration provided an opportunity to alter the service's execution path and obtain elevated privileges.

The attack was conducted from the attacker system through the isolated network, establishing remote access to the target, and leveraging the insecure file permissions to execute the modified service with elevated privileges. The resulting access was verified to determine whether the attack successfully achieved privilege escalation. Evidence was collected throughout the process to document the vulnerable configuration, remote access, and resulting elevated privilege level.

3.3 DETECTION & ESCALATION

Following the attack, the target workstation was monitored for evidence of unauthorized activity, and relevant Windows system and security events were

reviewed to identify indicators associated with the remote access and privilege escalation. The observed activity was correlated with the actions performed during the attack, allowing the compromise and its impact to be identified and documented. The incident was then escalated for further investigation and remediation based on the identified insecure file permissions, remote access, and elevated privileges.

3.4 REMEDIATION & VERIFICATION

The identified insecure file permission was remediated by restricting access to the affected service files so that unauthorized users could no longer modify them. The service configuration was then reviewed to confirm that the vulnerable permissions had been corrected. Verification testing was performed to ensure the previous attack path could no longer be used to obtain remote access or elevated privileges, confirming that the implemented remediation effectively reduced the identified security risk.

4.0 ATTACK & PRIVLEGE ESCALATION

4.1 INITIAL ACCESS

The attacker system was a Kali Linux virtual machine used to perform reconnaissance, authentication testing, and remote access against the Windows 10 target. SMB authentication was tested using controlled failed login attempts, which were later reviewed in the Windows Security logs to verify that the activity was being recorded.

Following the initial testing, remote access to the Windows workstation was established through RDP using the low-privileged labuser account. This provided a controlled low-privilege foothold on the target and established the starting point for the privilege-escalation assessment.

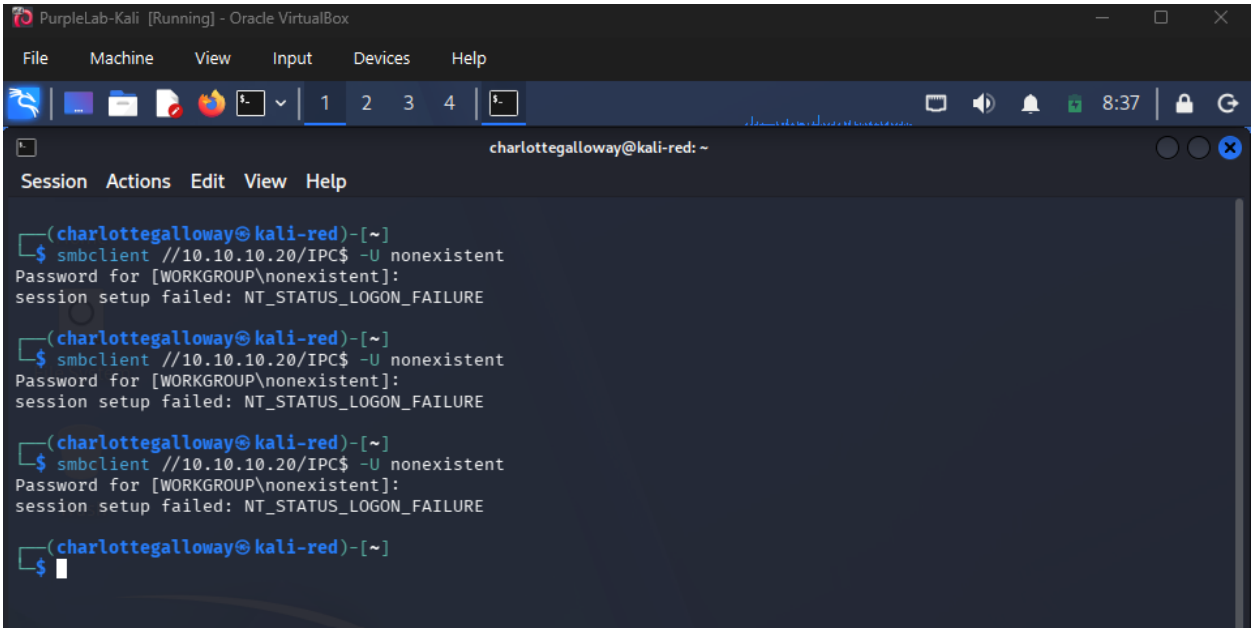

Figure 5. Controlled SMB authentication attempts generated from the Kali Linux attacker

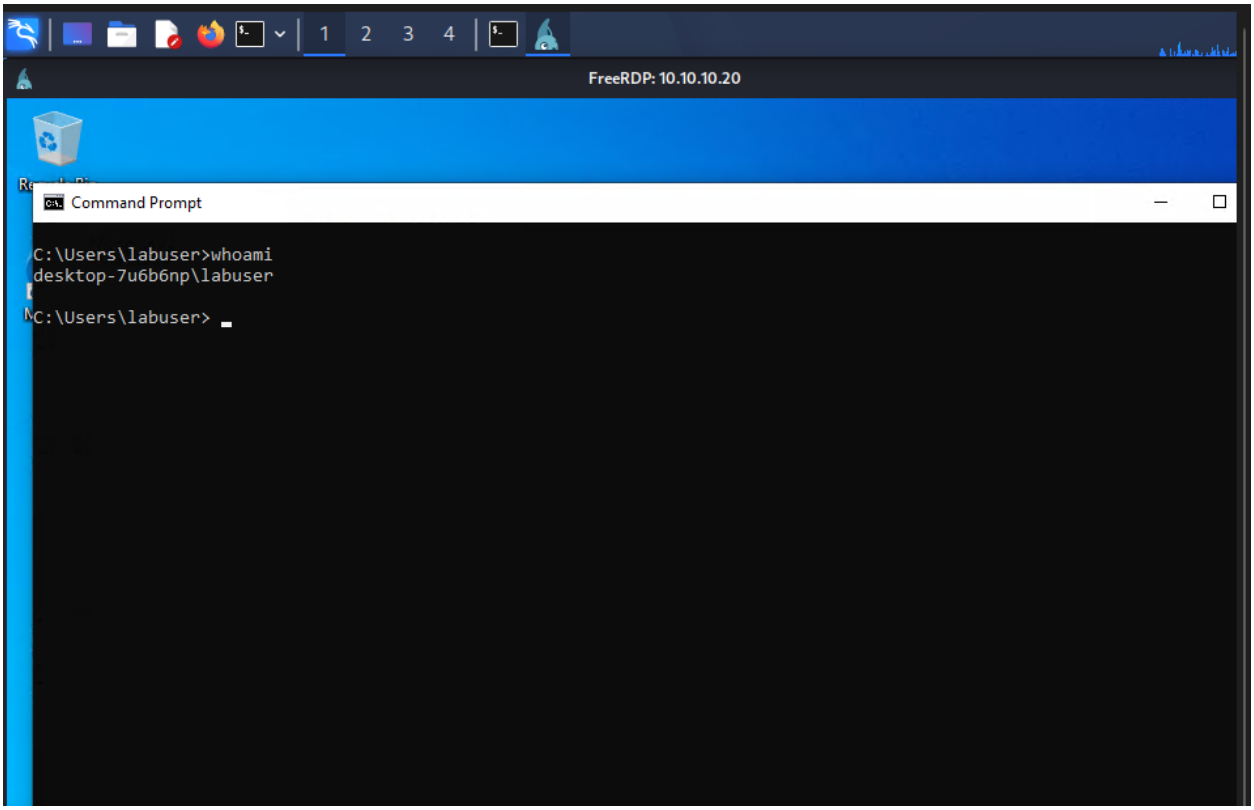

Figure 6. Remote Desktop Session established from Kali Linux to the low-privilege windows account.

4.2 PRIVLEGE ESCALATION

The privilege-escalation assessment focused on insecure file permissions associated with a Windows service configured to run as LocalSystem. The service executed C:\Lab\service.bat, while permissions on the file initially allowed lower-privileged users to modify it.

Because the service operated with SYSTEM-level privileges, the ability for a low-privileged user to modify a file executed by the service created a potential indirect path to privileged code execution. The escalation path was tested in the controlled lab environment; however, SYSTEM-level execution was not successfully demonstrated due to limitations in the service implementation.

The vulnerability was subsequently remediated by removing the unnecessary Modify permissions, and the fix was verified when labuser was no longer able to modify the service payload.

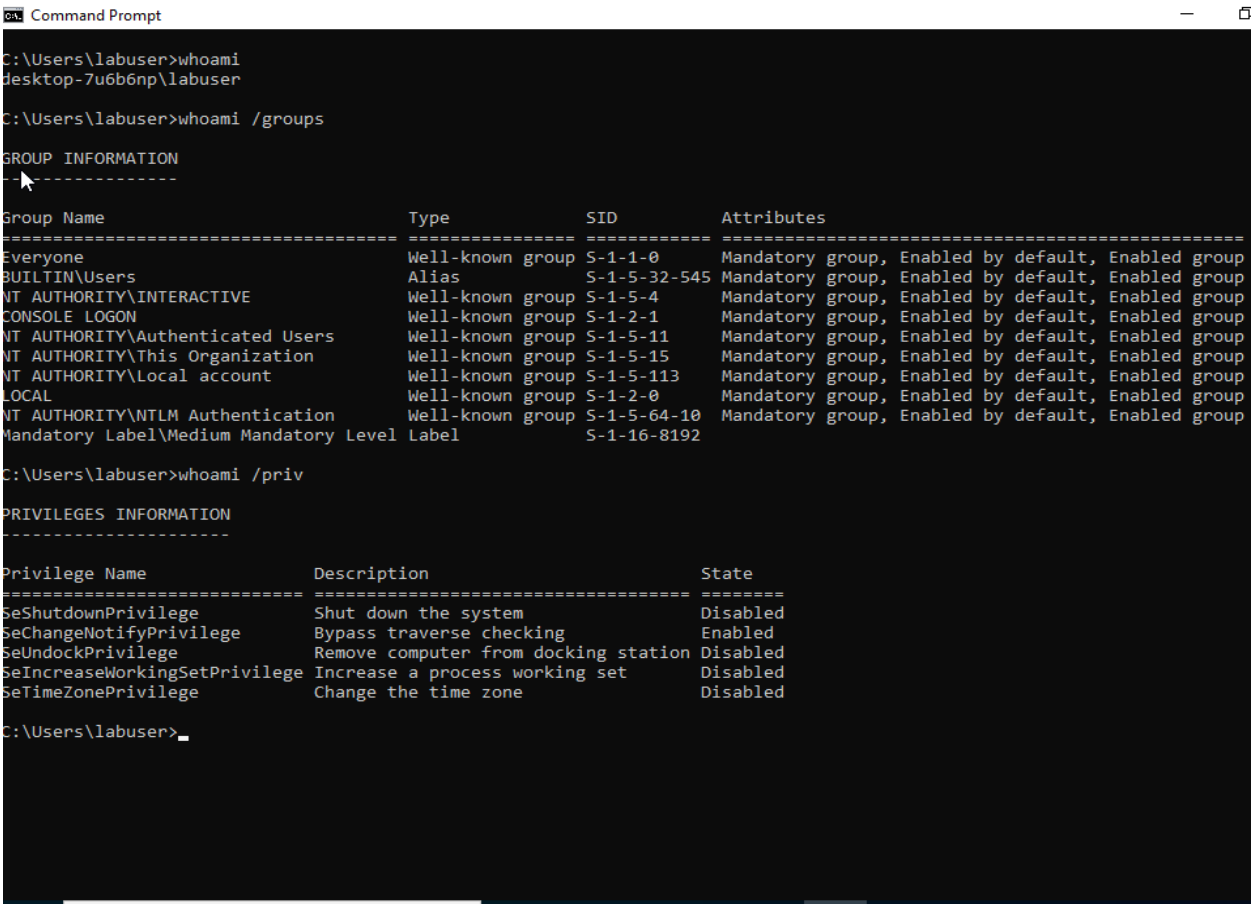

Figure 7. Low-privilege user baseline demonstrating labuser did not have admin privileges

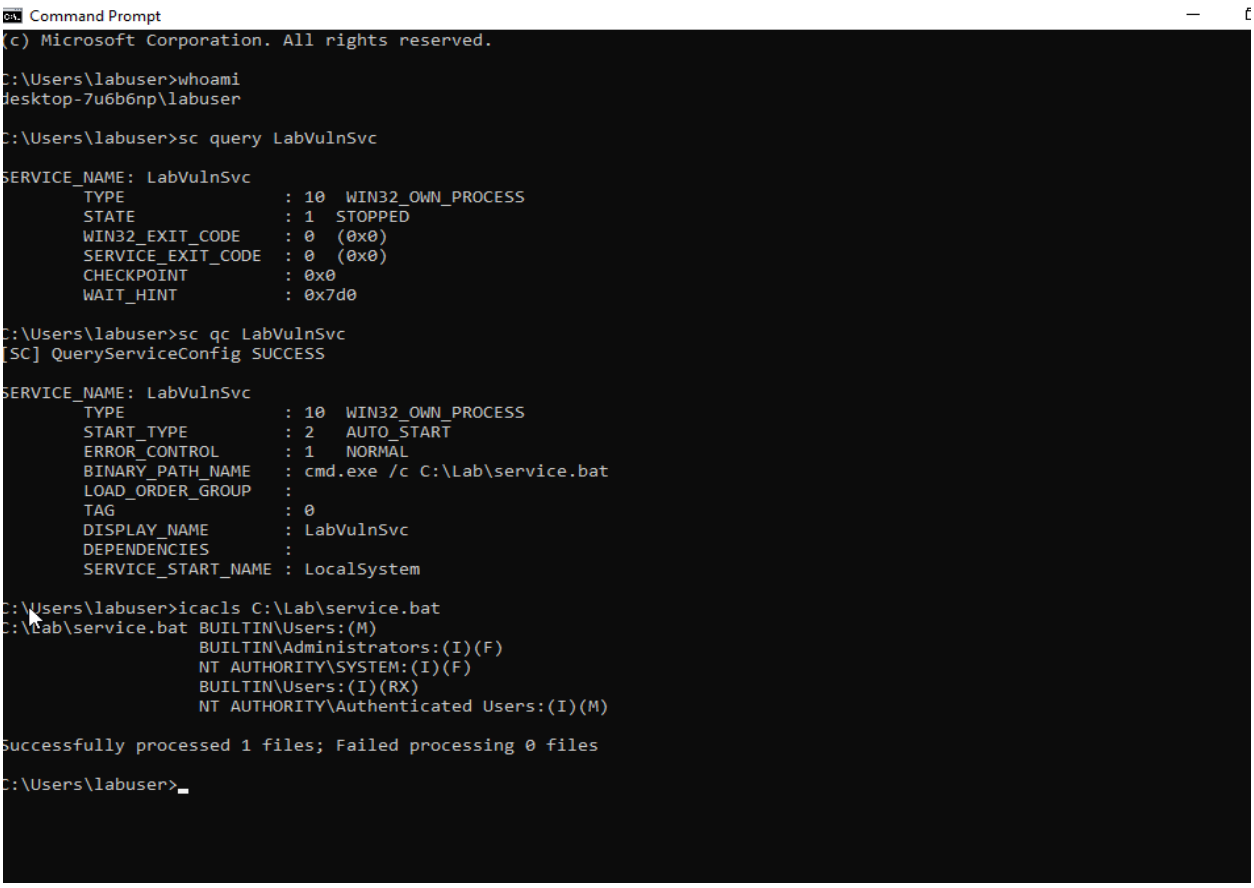

Figure 8. Attacker side discovery of the vulnerable service.

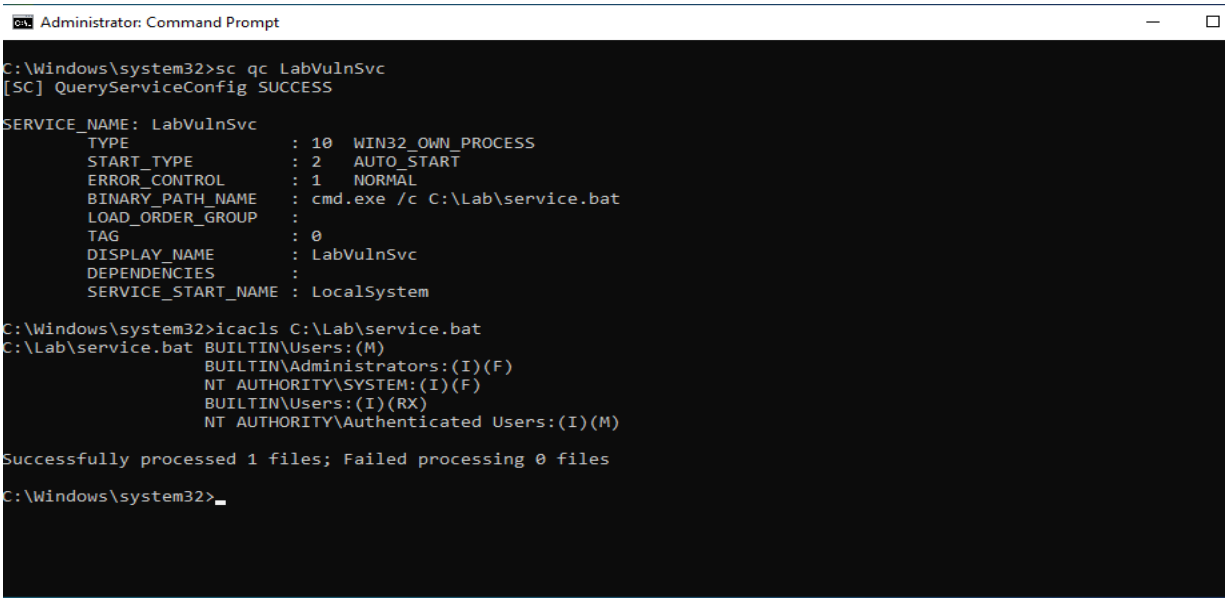

Figure 9. Vulnerable Service configuration showing LocalSystem execution and the insecure file permissions.

5.0 DETECTION & INVESTIGATION

5.1 WINDOWS SECURITY EVENTS

Windows Security Event Logs were reviewed to identify authentication activity generated during the assessment. Event ID 4625 recorded the controlled failed SMB authentication attempts originating from the Kali Linux attacker system at 10.10.10.10 The events showed a network logon type and unsuccessful authentication status, providing evidence of the simulated attack activity.

Event ID 4624 was also identified for the successful RDP connection using the labuser account. The event recorded Logon Type 10 and the Kali Linux source address, allowing the successful remote session to be correlated with the attack activity.

Together, these events demonstrated that Windows Security logs provided useful telemetry for identifying both failed authentication attempts and successful remote access.

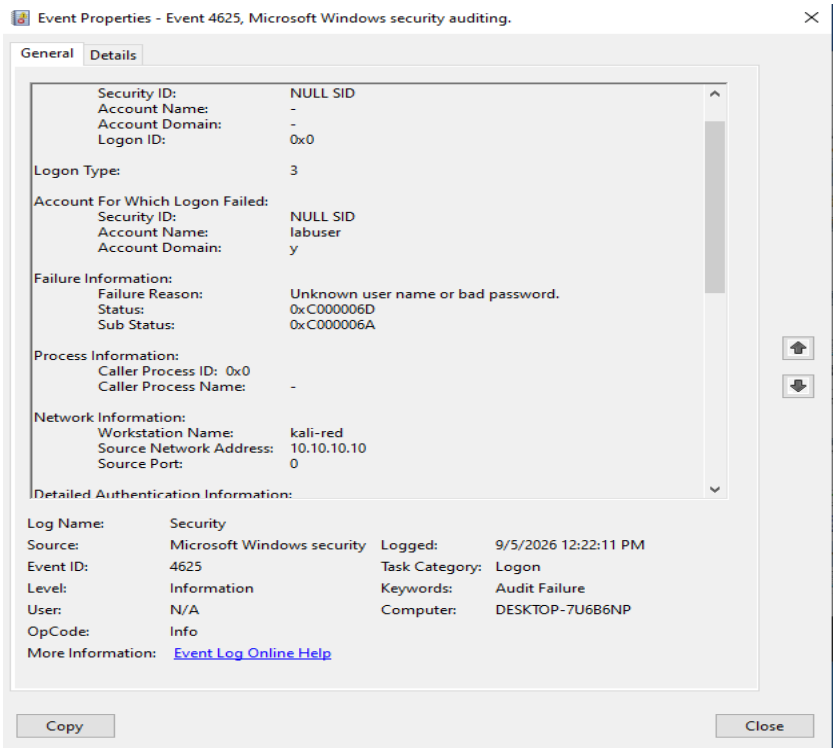

Figure 10. Windows Security event 4625 documenting failed SMB authentication from attacker 'kali-red'.

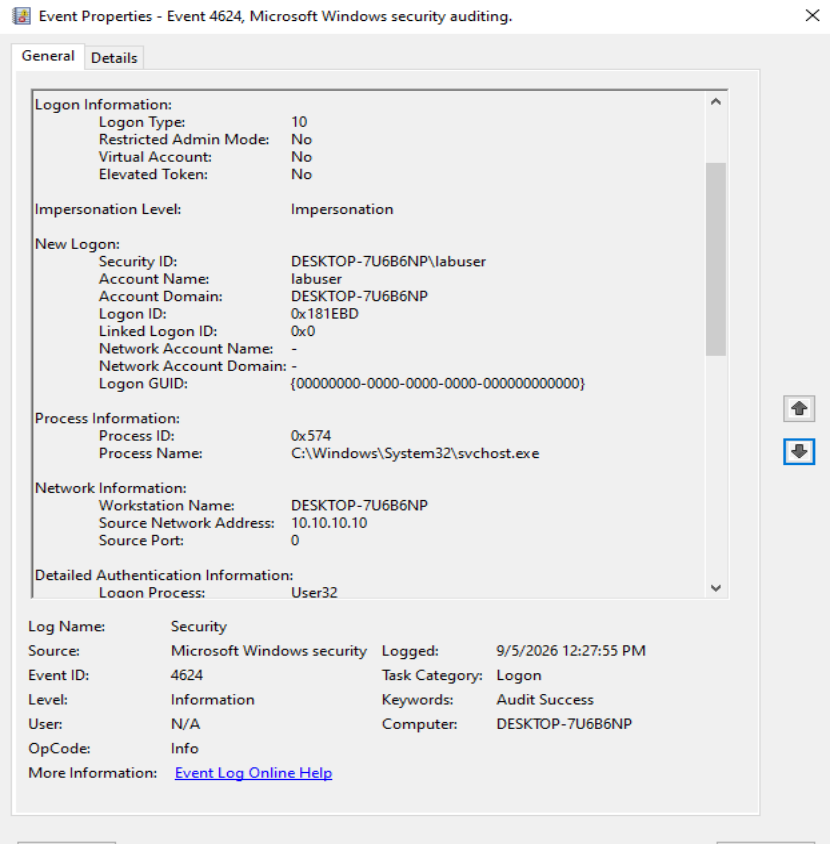

Figure 11. Windows security log event 4624 documenting successful RDP logon from attacker source address.

5.2 NETWORK ANALYSIS

Wireshark was used to analyze network traffic between the Kali Linux attacker system and the Windows 10 target. The capture identified communication between 10.10.10.10 and 10.10.10.20, including TCP traffic associated with Remote Desktop Protocol on port 3389.

The RDP traffic was protected by TLS, meaning the contents of the remote session were not visible as plaintext in the packet capture. However, network-level information such as source and destination addresses, ports, protocols, and connection activity remained observable. This demonstrated how network monitoring can provide useful evidence of remote access even when application data is encrypted.

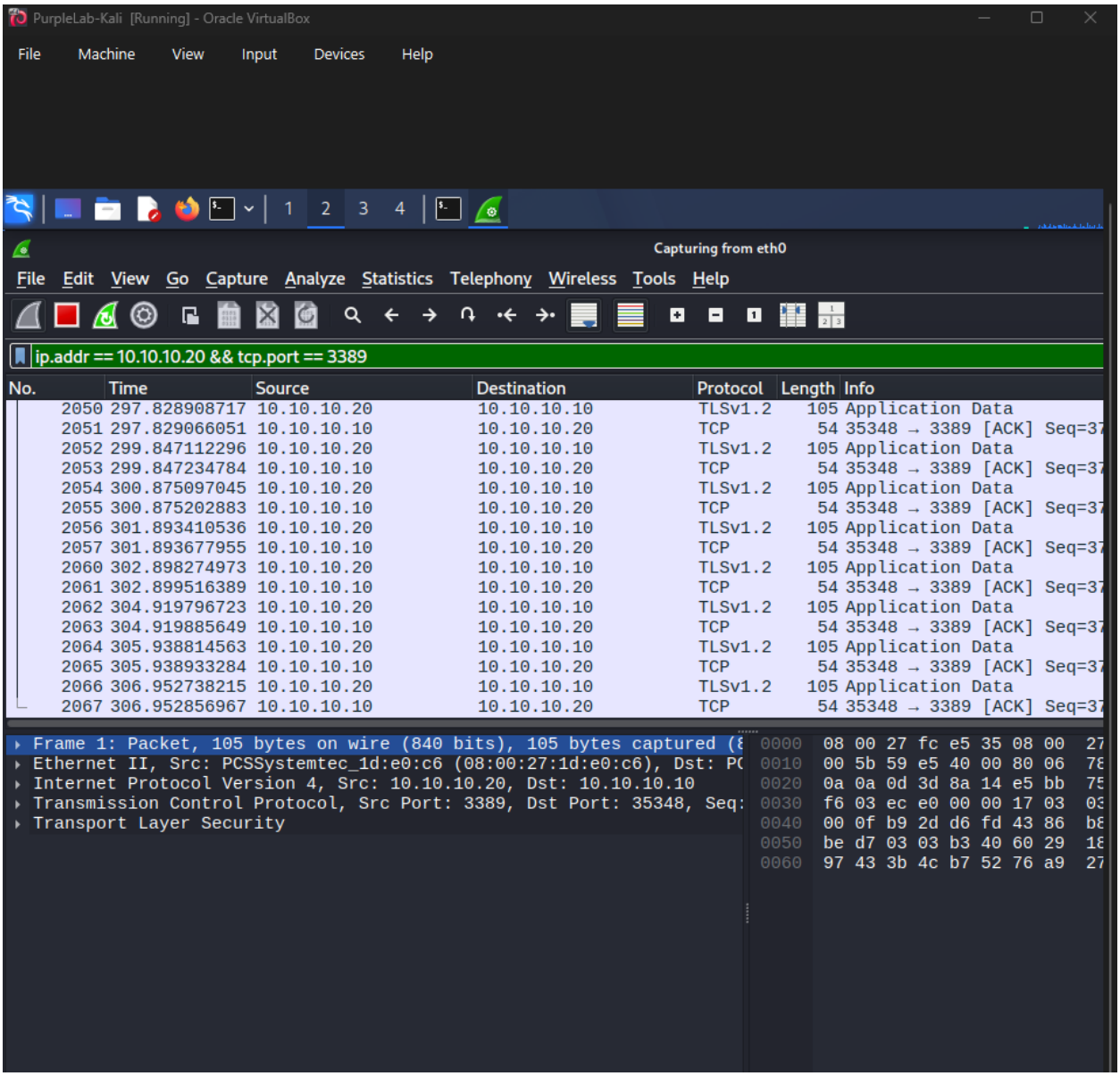

Figure 12. Wireshark capture showing encrypted RDP traffic between attacker and target.

6.0 REMEDIATION & VERIFICATION

6.1 REMEDIATION

The identified privilege-escalation condition was remediated by removing unnecessary Modify permissions from the service payload and its inherited permissions. The C:\Lab\service.bat file was restricted so that low-privileged users retained read and execute access but could no longer modify the file.

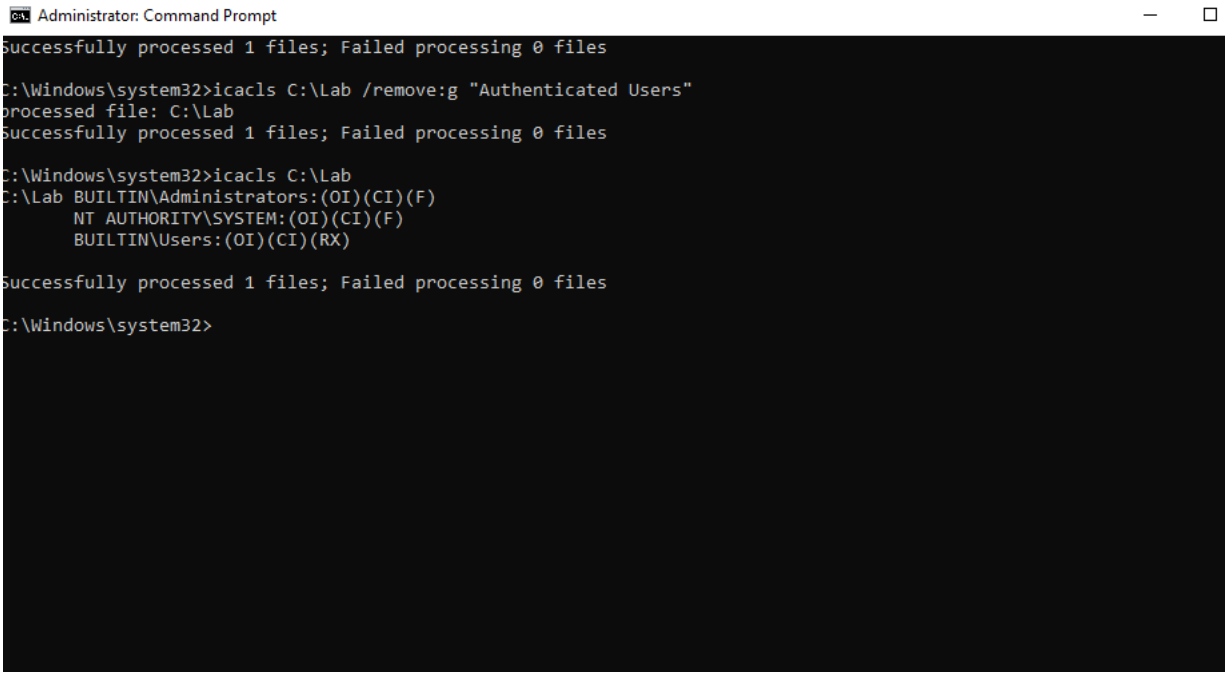

Figure 13. Remediation of Modify permissions of low-privilege user.

6.2 VERIFICATION

The remediation was verified by reconnecting as the low-privileged labuser account and attempting to modify the service payload. The modification attempt returned “Access is denied,” confirming that the user could no longer write to the file executed by the LocalSystem service.

This verification demonstrated that the identified file-permission vulnerability had been successfully mitigated.

Figure 14. Verification that the low-privilege account can no longer modify the service payload.

7.0 FINDINGS & LESSON LEARNED

The assessment identified insecure file permissions associated with a vulnerable service as the primary security weakness. The misconfiguration allowed unauthorized modification of service-related files, which provided an attack path for remote access and privilege escalation. The exercise demonstrated the importance of applying least-privilege principles to service files and regularly reviewing permissions on critical system resources. It also showed that effective detection requires correlating Windows security events with network activity to identify suspicious behavior and understand the progression of an attack. Finally, the remediation and verification process demonstrated that correcting the underlying permission weakness and retesting the attack path are necessary to confirm that a vulnerability has been effectively mitigated.

8.0 CONCLUSION

This Purple Team lab demonstrated the interaction between offensive security activity and defensive detection within an isolated Windows environment. The assessment progressed from network reconnaissance and controlled authentication testing to low-privilege remote access and the identification of an insecure service file-permission configuration that created a potential path to SYSTEM-level execution.

Windows Security logs and network traffic analysis provided visibility into the simulated attack activity, while the identified permission vulnerability was successfully remediated and verified. Although SYSTEM-level execution was not successfully achieved due to limitations in the service implementation, the lab demonstrated the complete security workflow of identifying a vulnerability, analyzing its potential impact, applying remediation, and validating the result.

Overall, the assessment reinforced the importance of least-privilege access, secure service configuration, centralized logging, and verification following remediation.