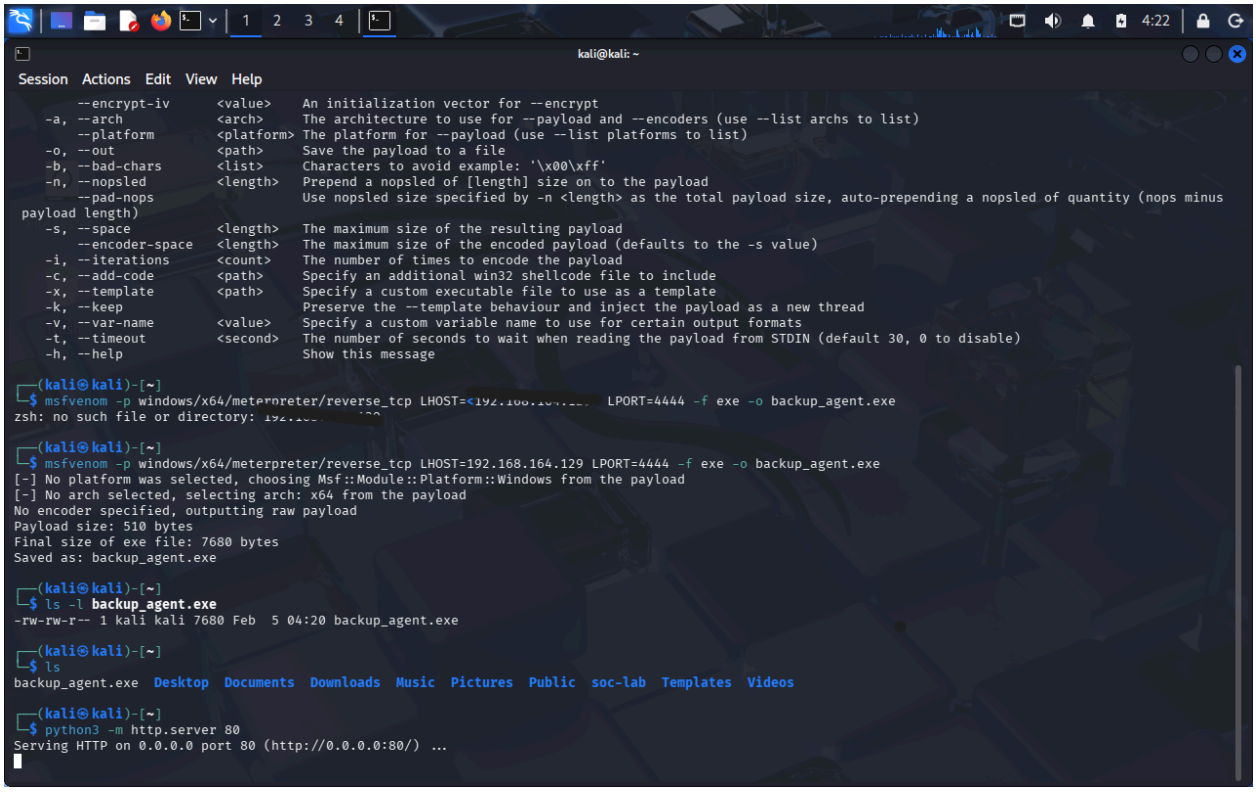
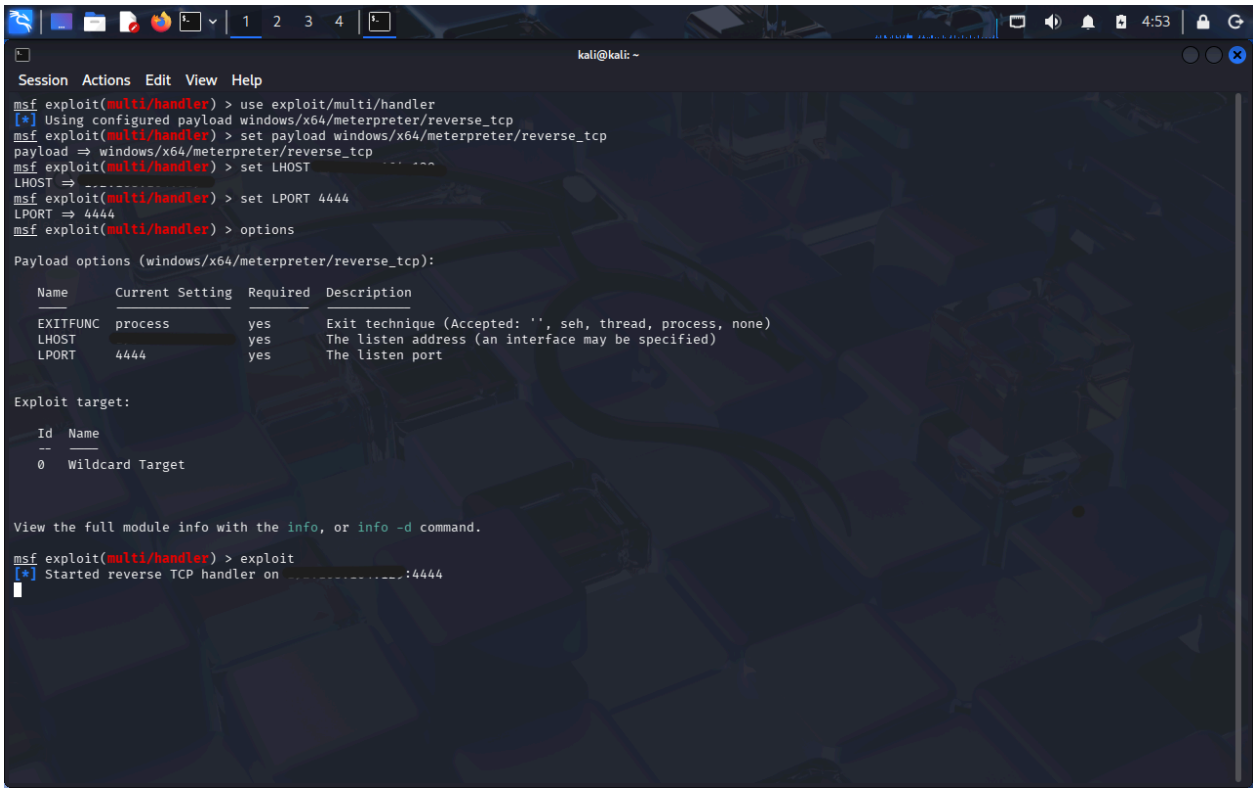
SOC Incident Report

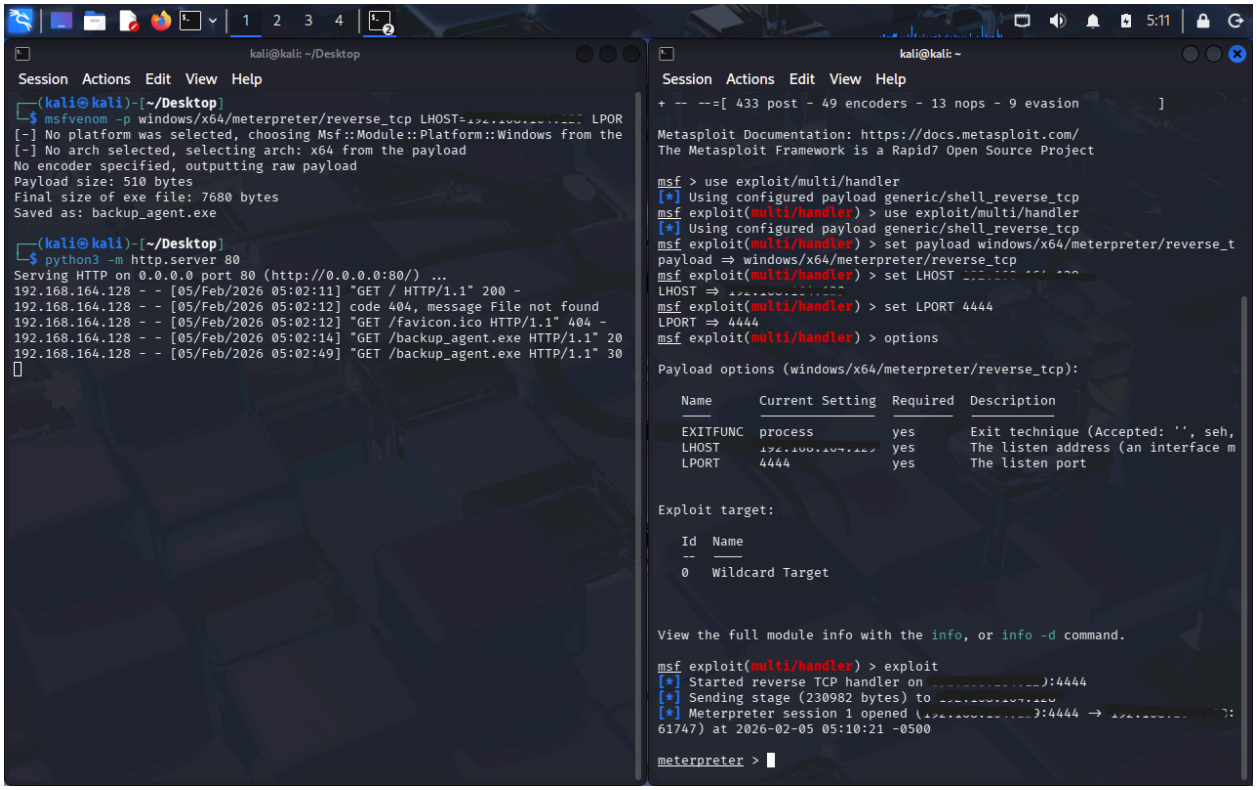
Active Attack Simulation & Forensic Investigation (Proof)

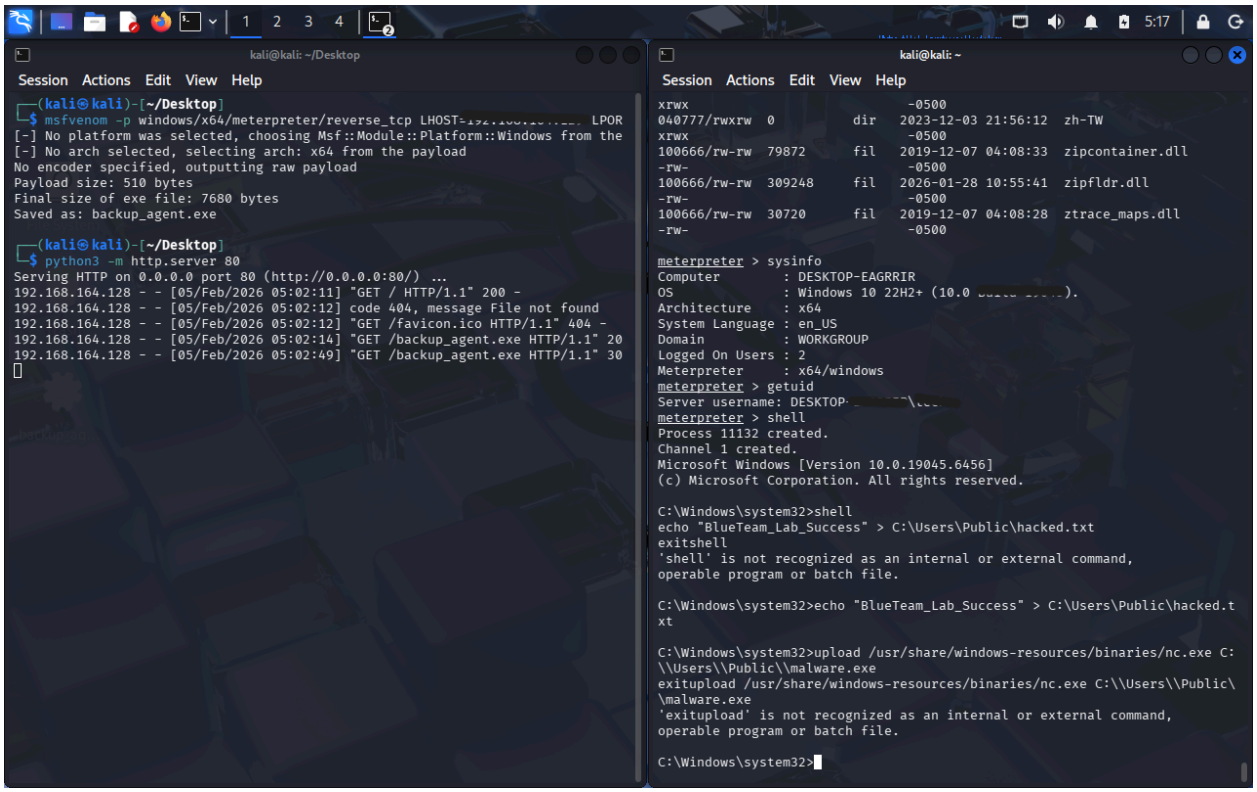
Kali: Attacker side proof



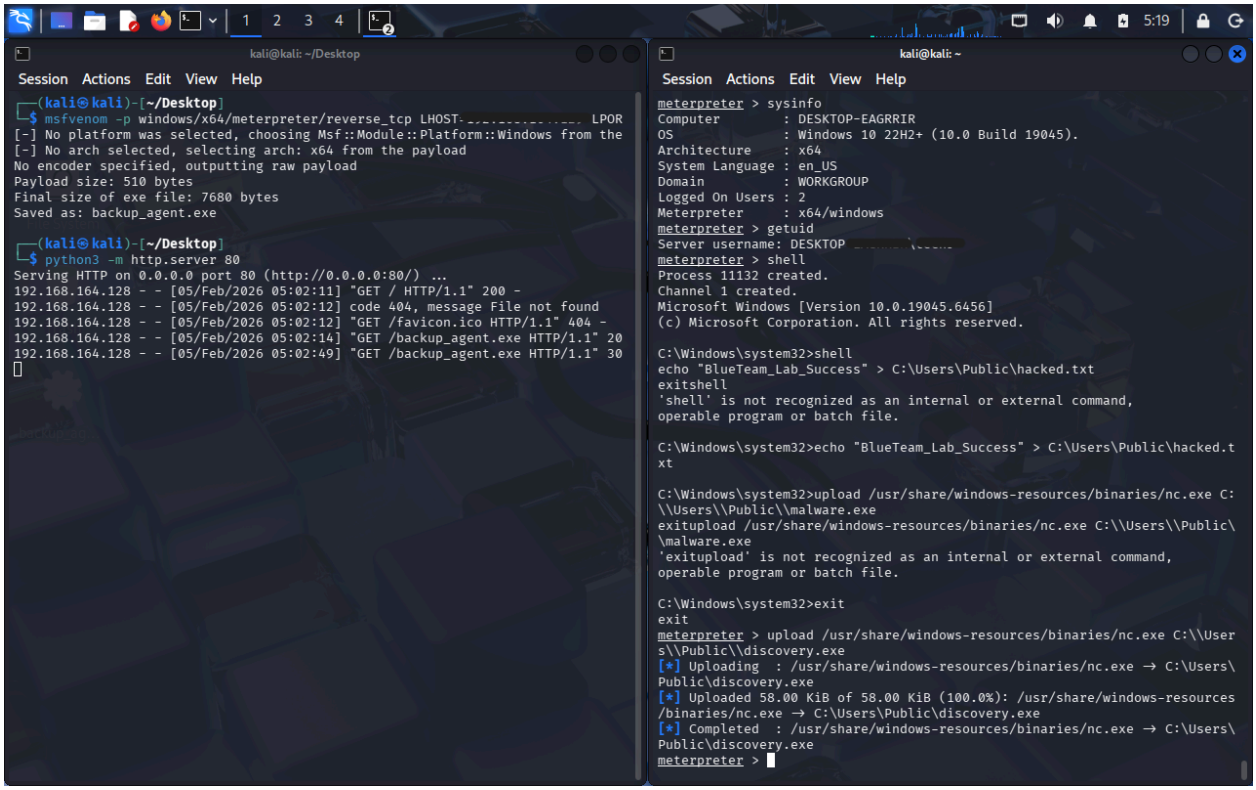


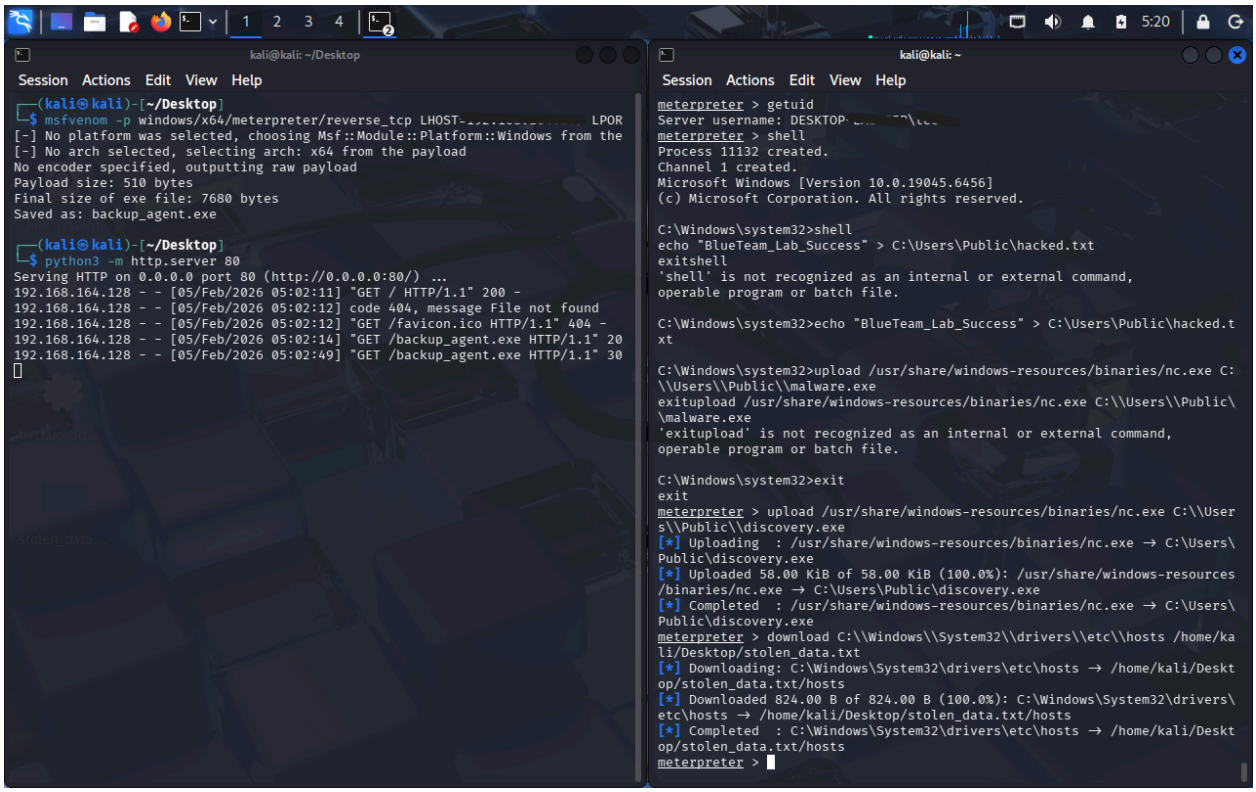
SOC Incident Report



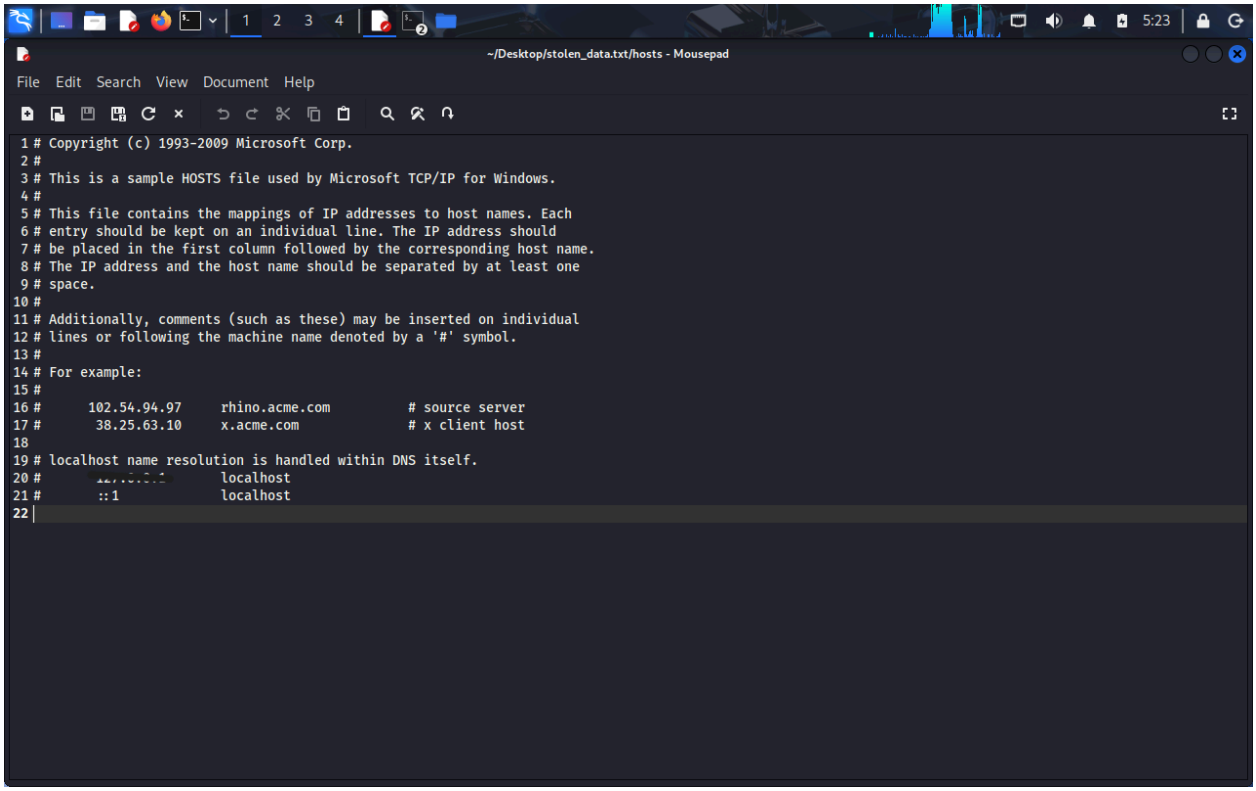


SOC Incident Report

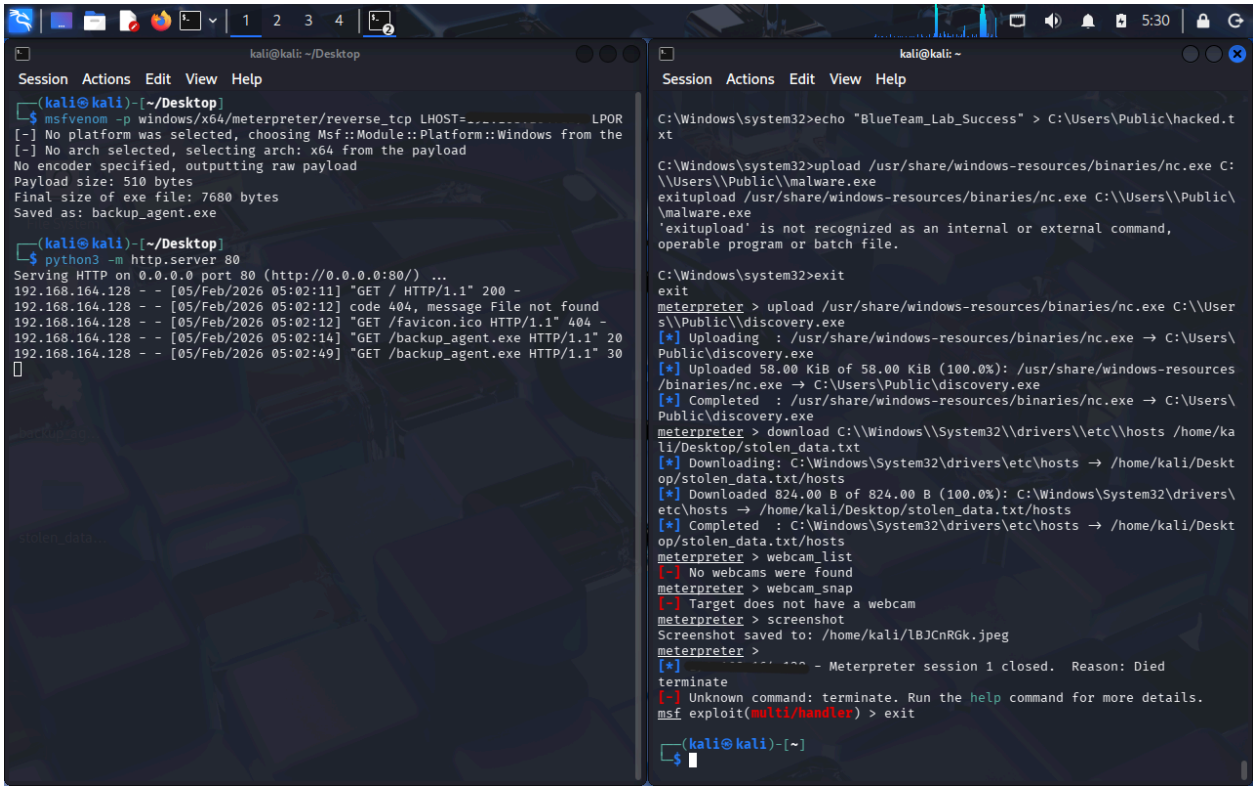


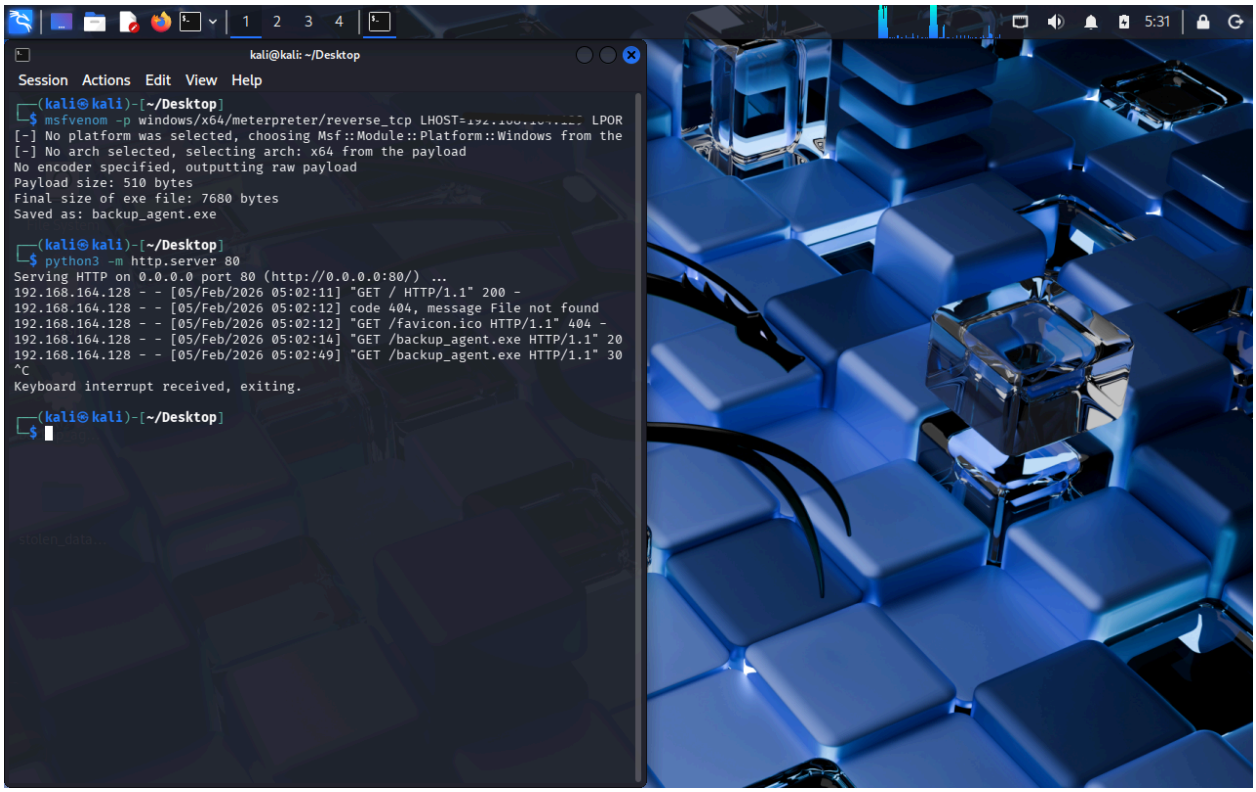


SOC Incident Report

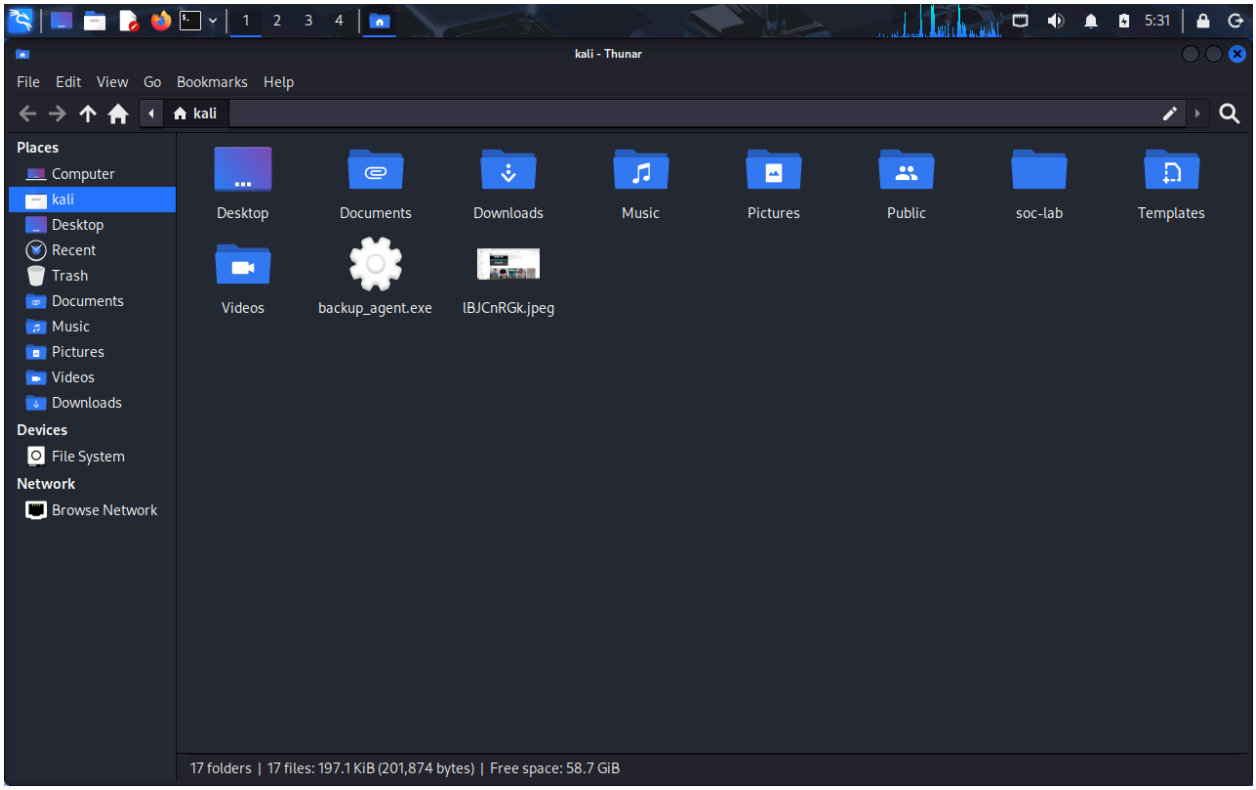


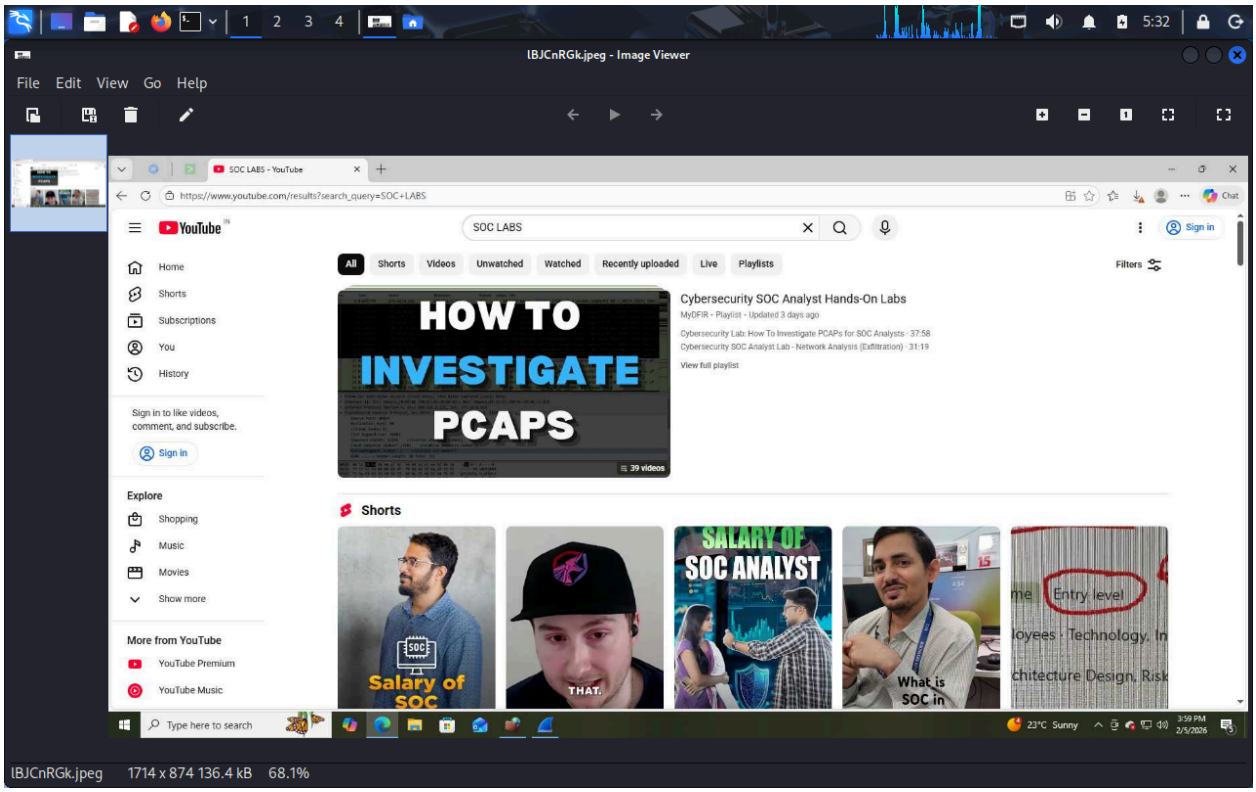
SOC Incident Report





SOC Incident Report





Active Attack Simulation & Forensic Investigation – Complete Attack Flow

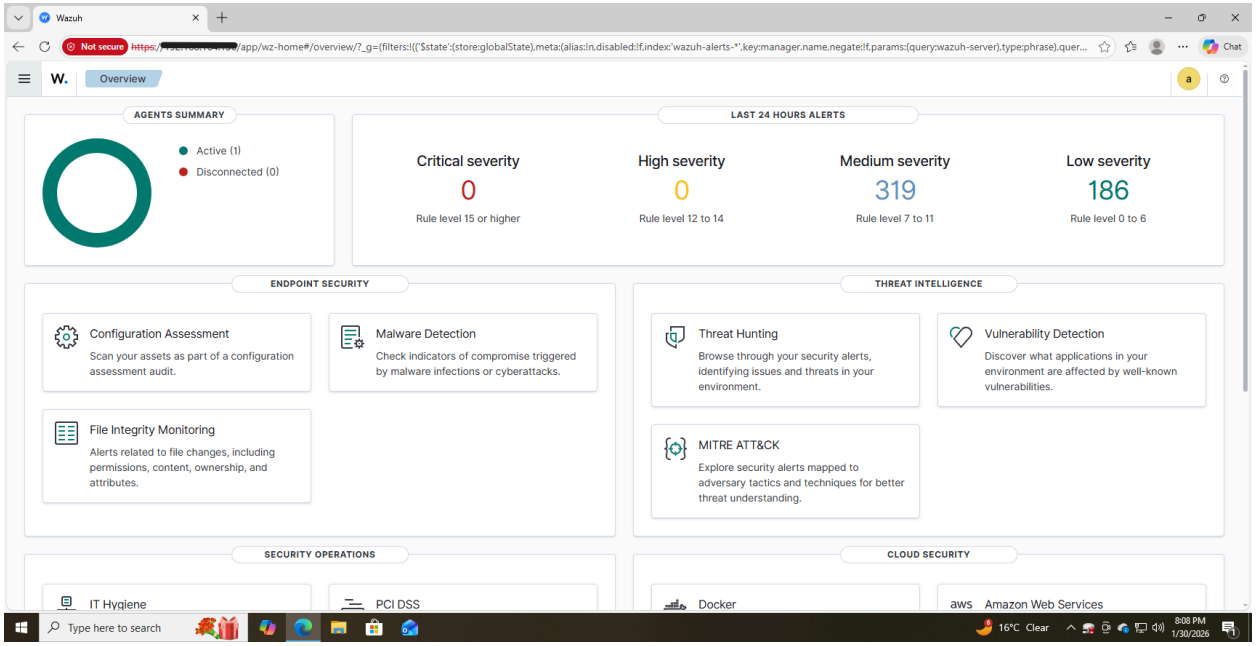
- Kali Linux (Attacker VM) created a malicious payload using `msfvenom` (windows/x64/meterpreter/reverse_tcp) and generated an executable file named `backup_agent.exe`.
- A Python HTTP server was started on Kali (port 80) to host the malicious executable for delivery to the target system.
- On Windows 10 (Victim VM), the payload file `backup_agent.exe` was downloaded from the Kali machine.
- Metasploit `multi/handler` was configured on Kali with:
    - Payload: windows/x64/meterpreter/reverse_tcp
    - LHOST: Attacker VM IP
    - LPORT: 4444
- The handler was started and began listening for incoming reverse connections.
- When the victim executed `backup_agent.exe`, a reverse TCP connection was established from the Windows VM to the Kali VM on port 4444.
- A Meterpreter session was successfully opened, confirming remote access to the Windows VM.
- Post-exploitation enumeration was performed:
    - `sysinfo` confirmed Windows 10 22H2 x64 environment.
    - `getuid` identified the logged-in user context.
- An interactive shell was spawned from Meterpreter.
- A proof-of-compromise file was created on the victim:
    - `echo "BlueTeam_Lab_Success" > C:\Users\Public\hacked.txt`
- A binary (`nc.exe`) was uploaded to the victim system as `discovery.exe`, simulating tool transfer during lateral movement or persistence staging.
- The Windows `hosts` file was downloaded from:
    - `C:\Windows\System32\drivers\etc\hosts`
    - Saved on Kali under `stolen_data.txt`
    - Demonstrates data exfiltration capability.
- Attempted webcam access commands showed no webcam present (post-exploitation capability testing).
- A screenshot of the victim desktop was captured using Meterpreter.
- The Meterpreter session was terminated.
- On Windows Event Viewer:
    - Event ID 5156 (Filtering Platform Connection) logged permitted outbound connection to attacker IP on port 4444.
    - Event ID 1001 (Windows Error Reporting) logged application crash related to `backup_agent.exe`.

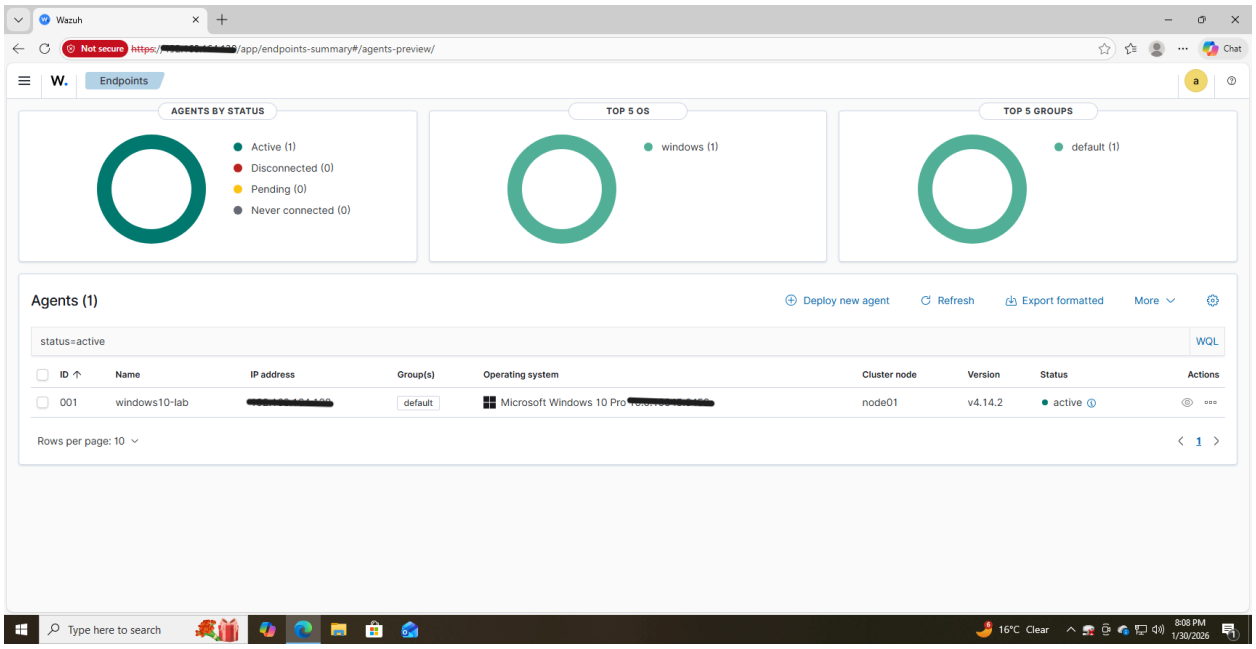
- Registry inspection showed UserAssist entries indicating execution artifacts of the malicious executable.
- Evidence artifacts generated:
    - Malicious executable on victim
    - Reverse connection logs
    - Security event logs
    - Application crash logs
    - Registry execution traces
    - Downloaded system file (hosts)
    - Desktop screenshot

This demonstrates a full attack lifecycle:
Payload Generation → Delivery → Execution → Command & Control → Post-Exploitation → Evidence Collection → Forensic Validation.

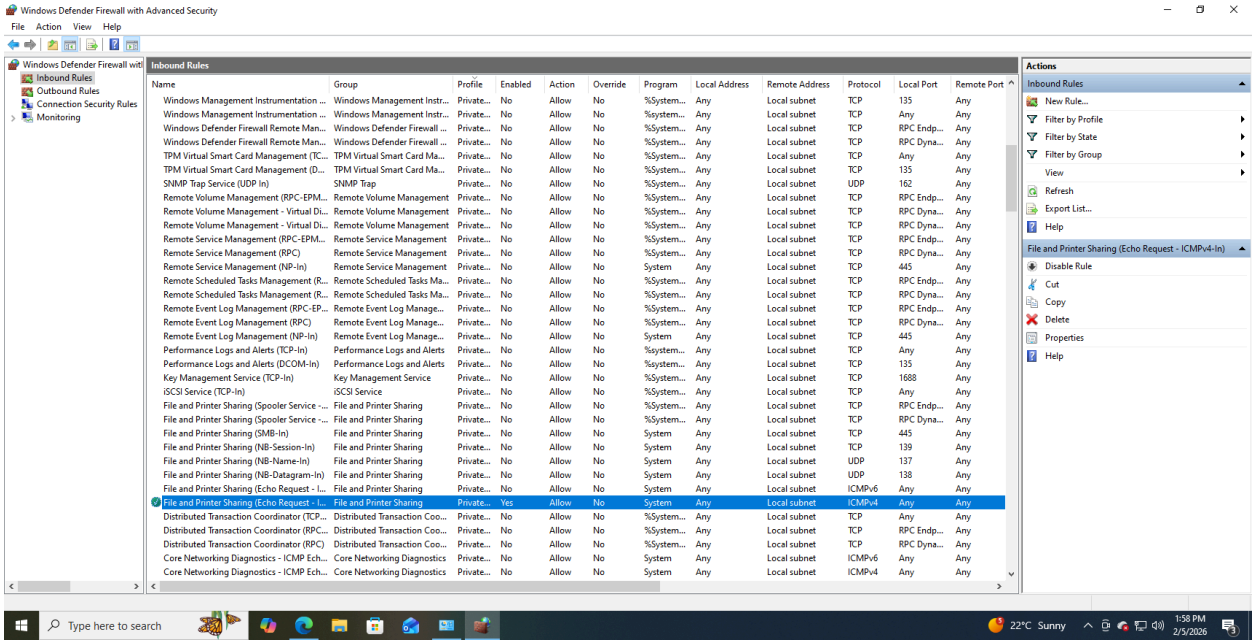
Windows: Defensive Side Investigation

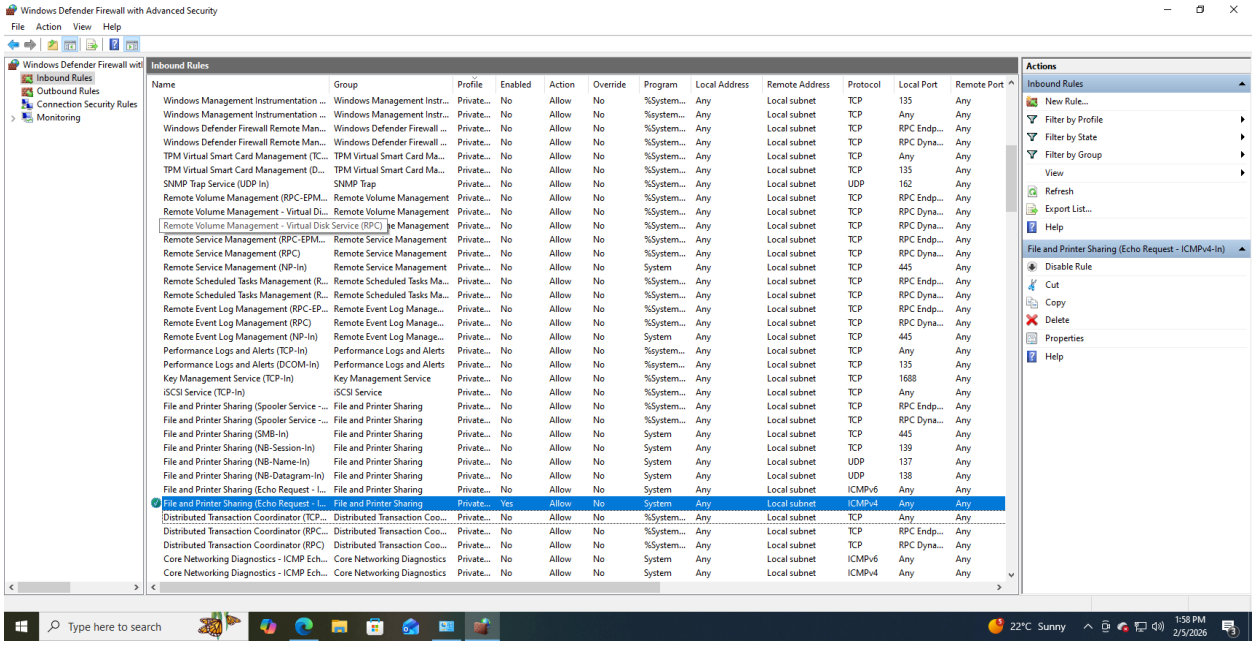
SOC Incident Report



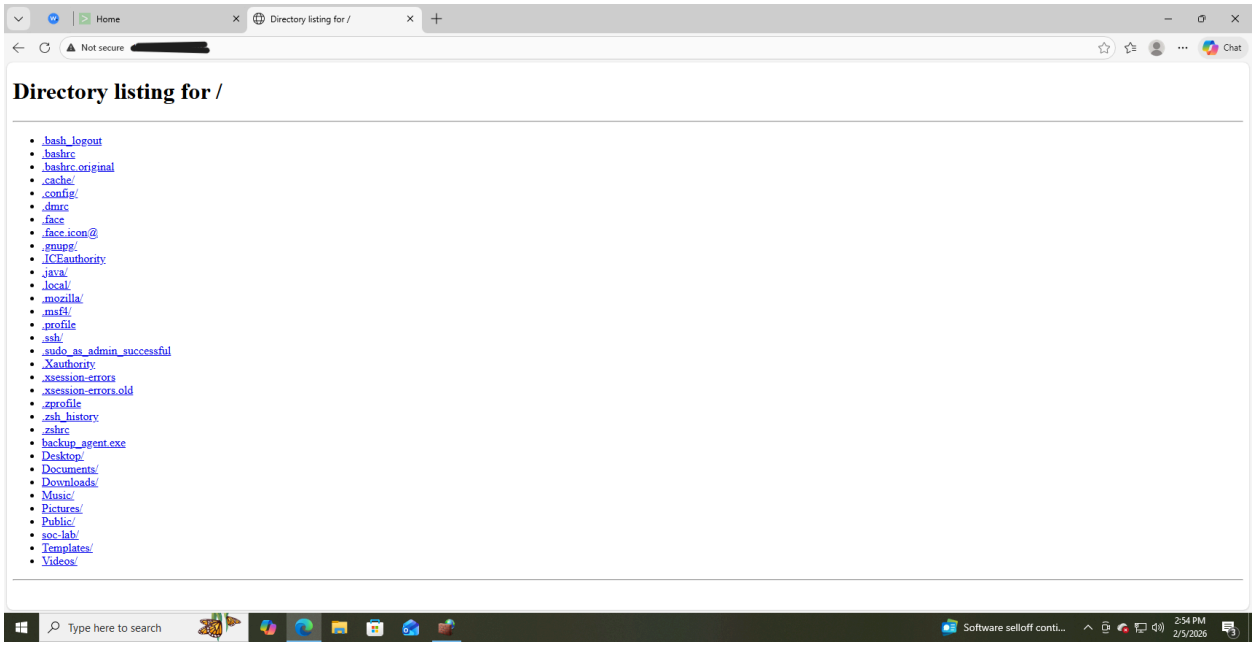


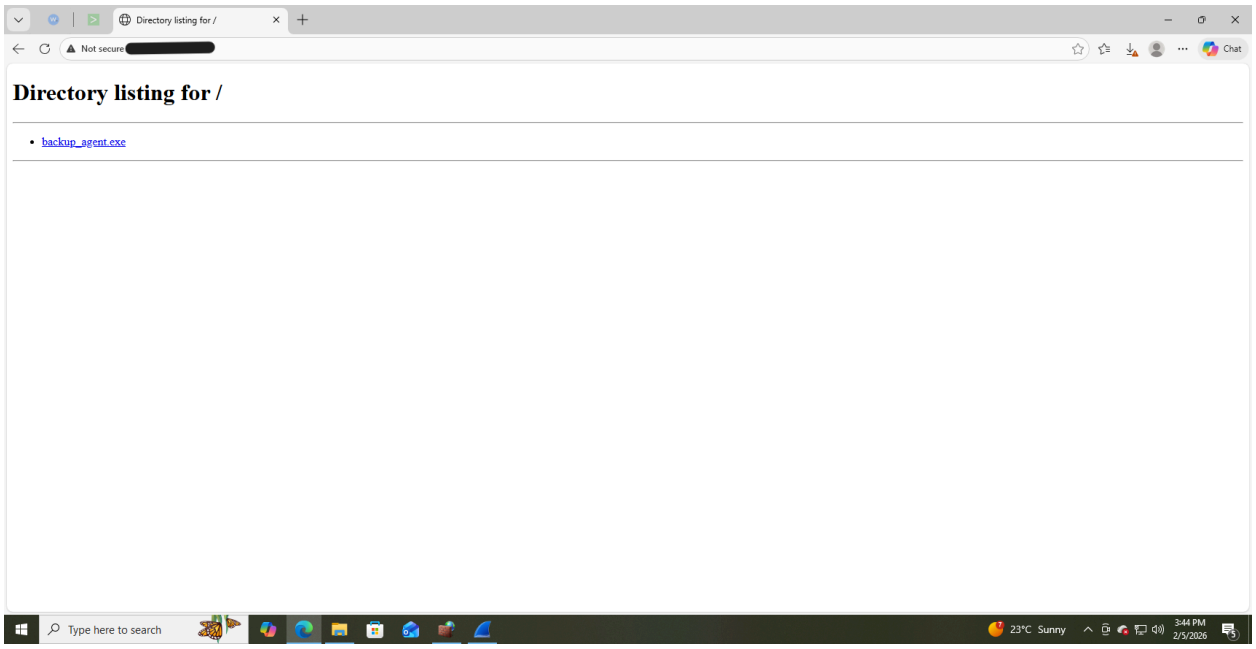
SOC Incident Report



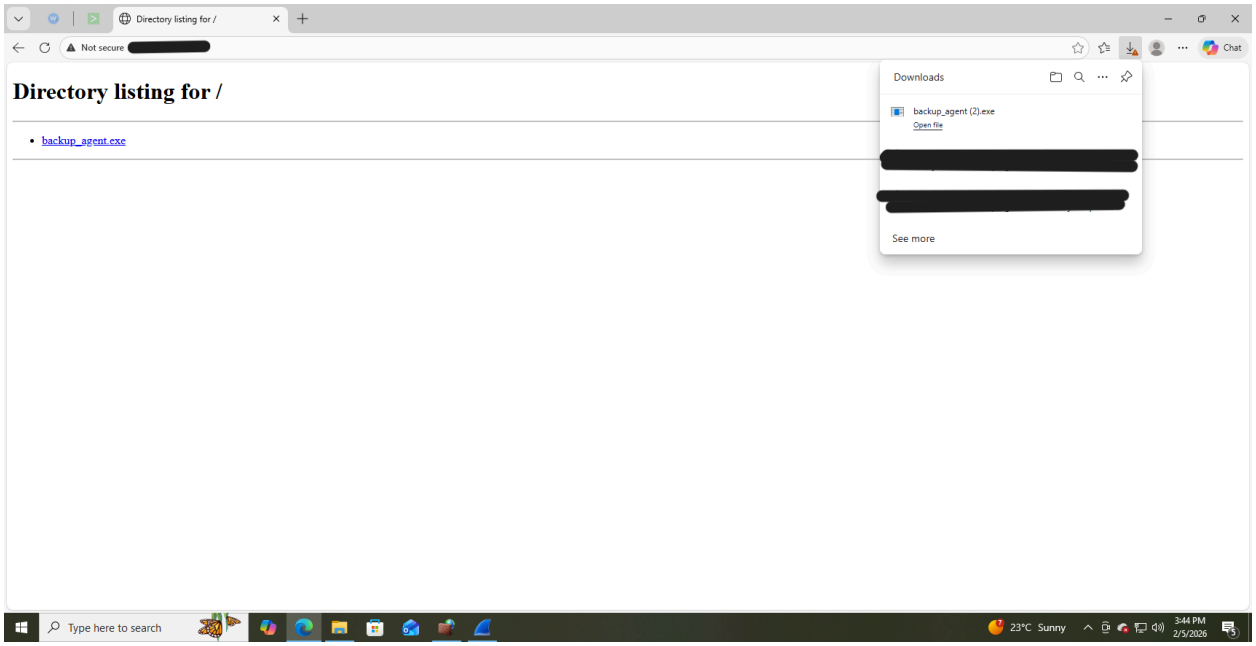


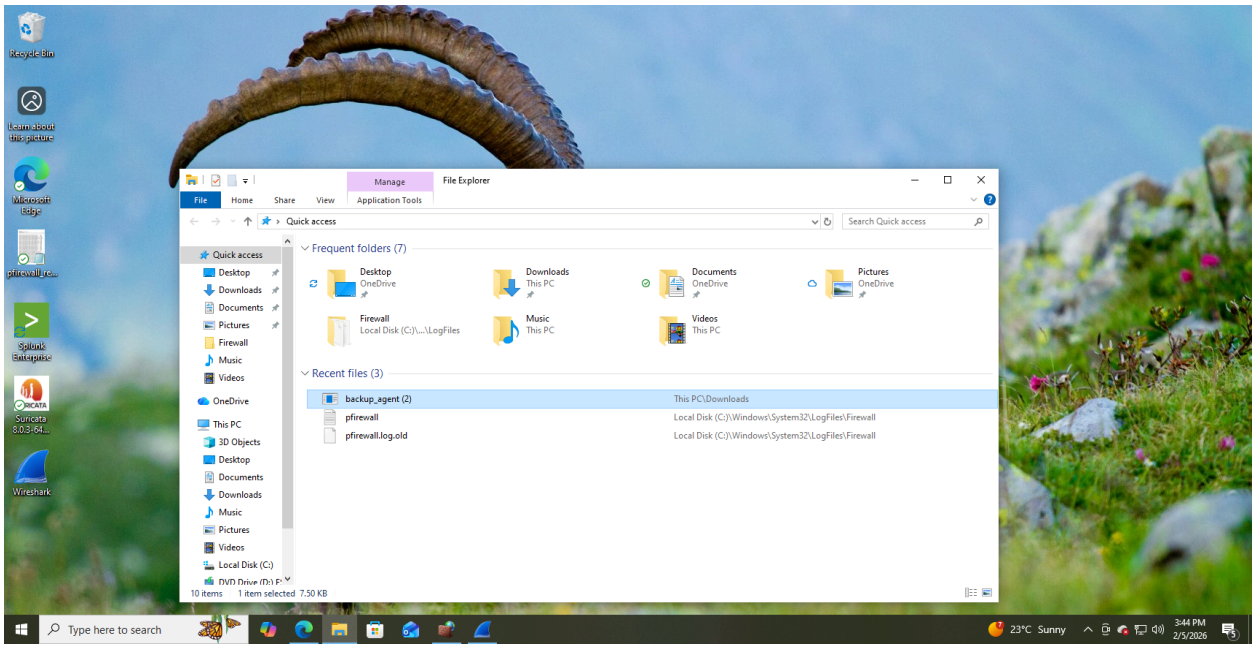
SOC Incident Report



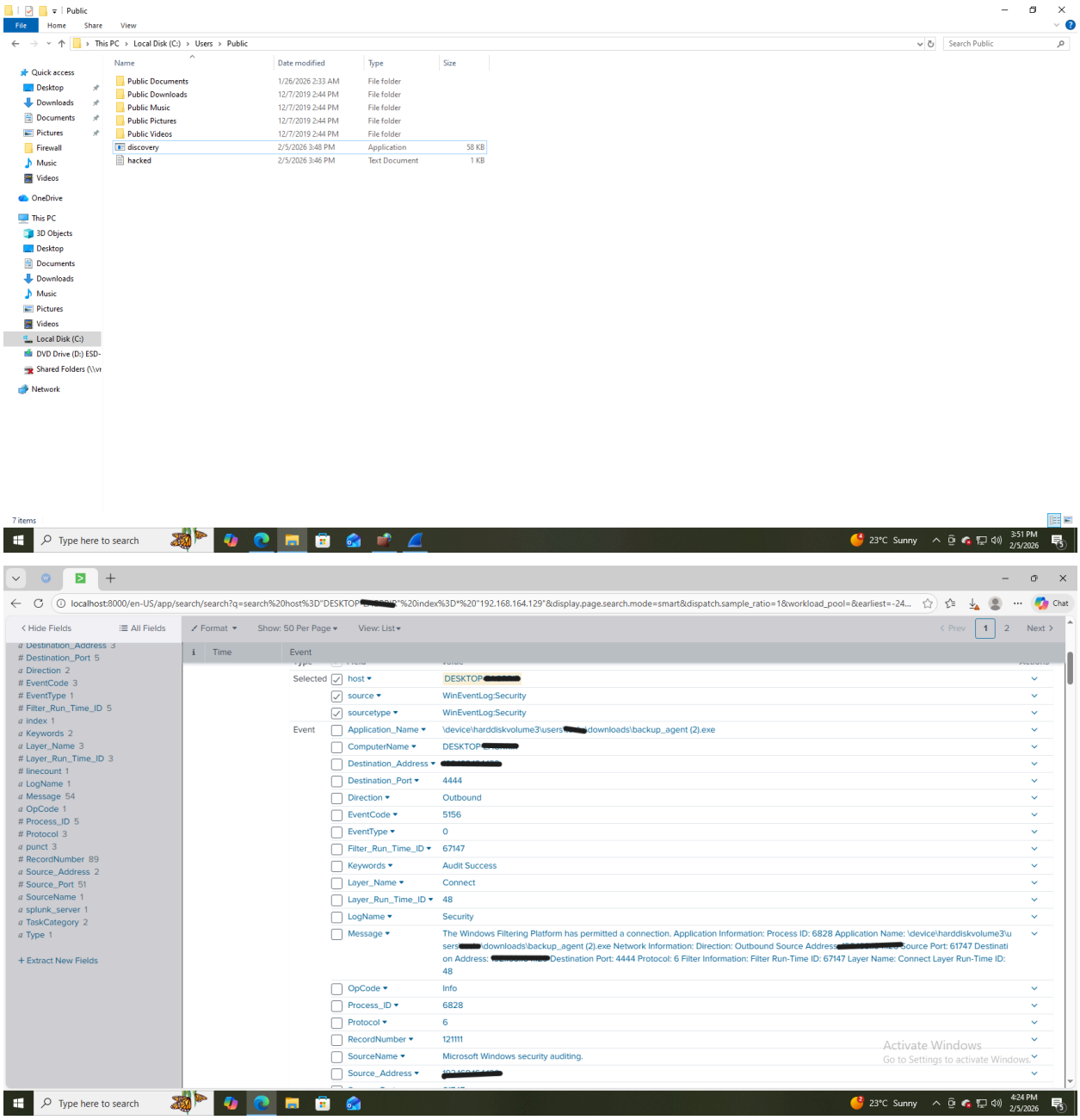


SOC Incident Report

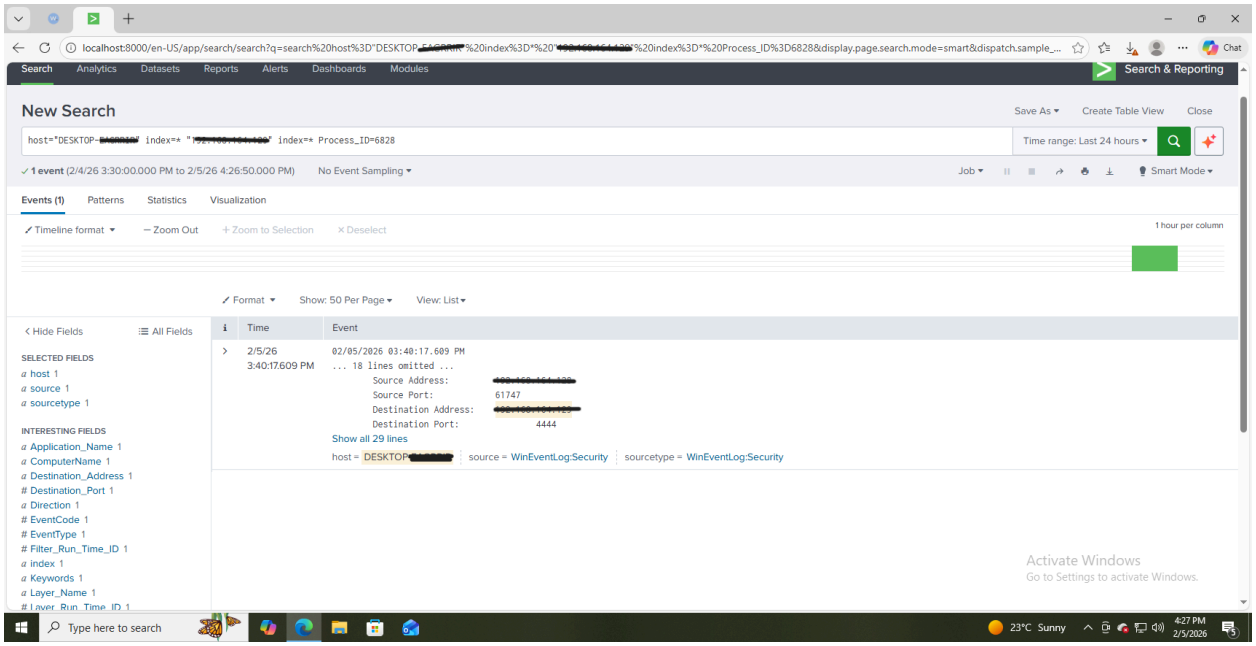


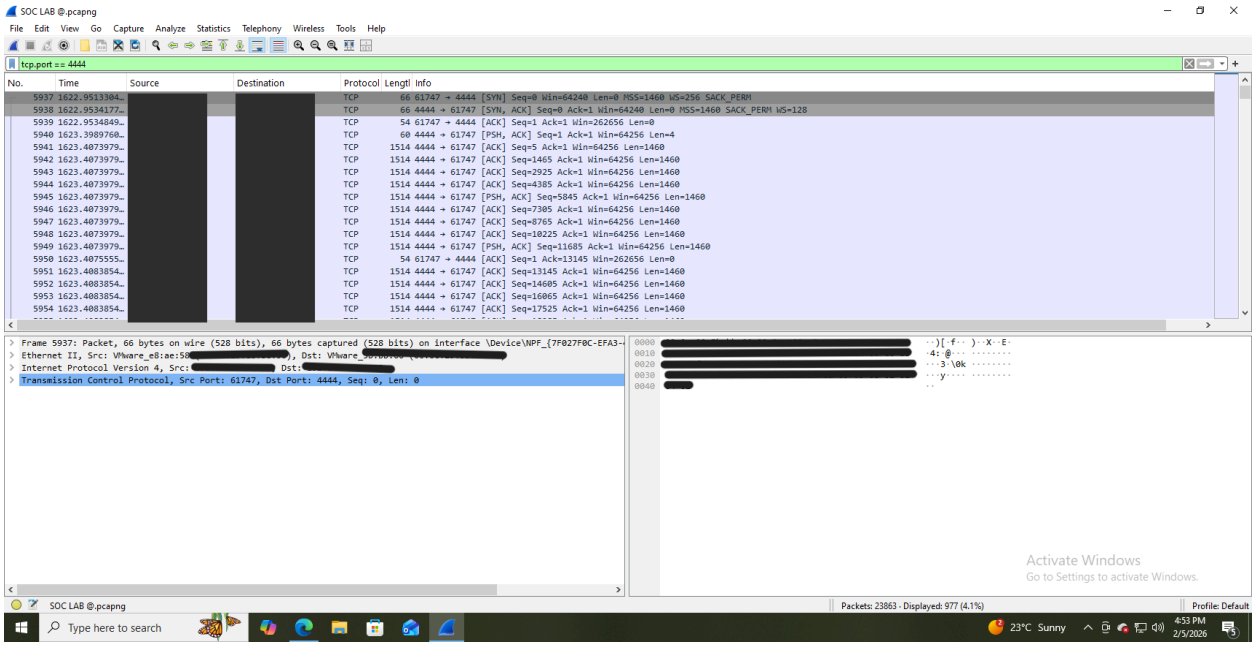


SOC Incident Report

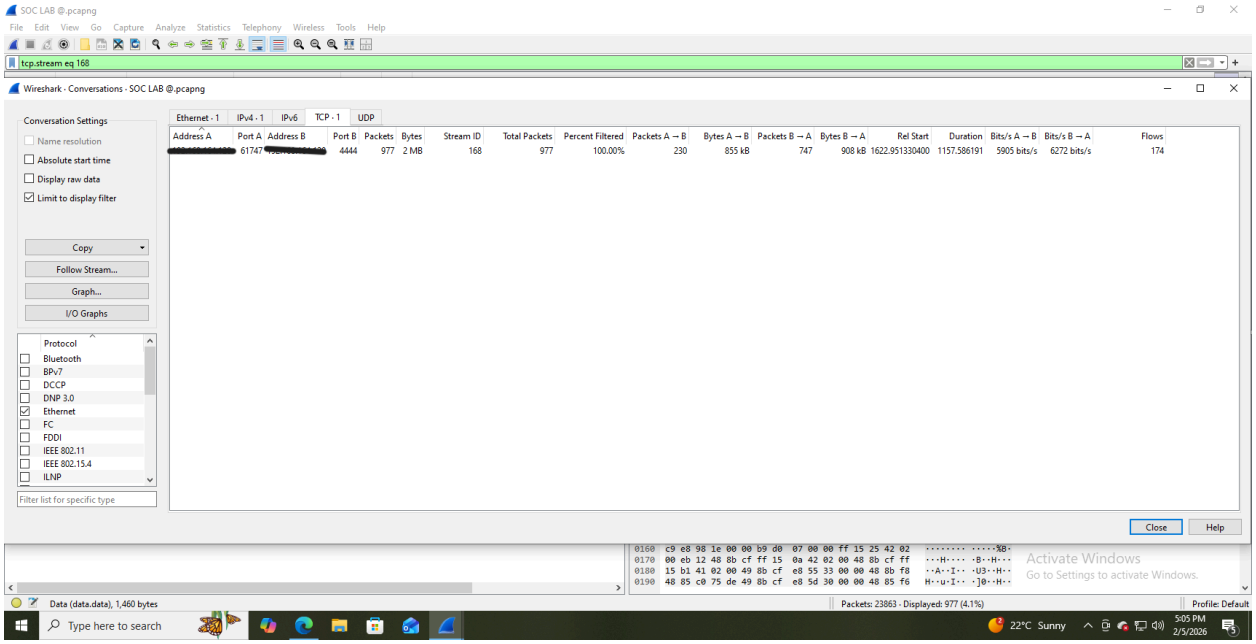


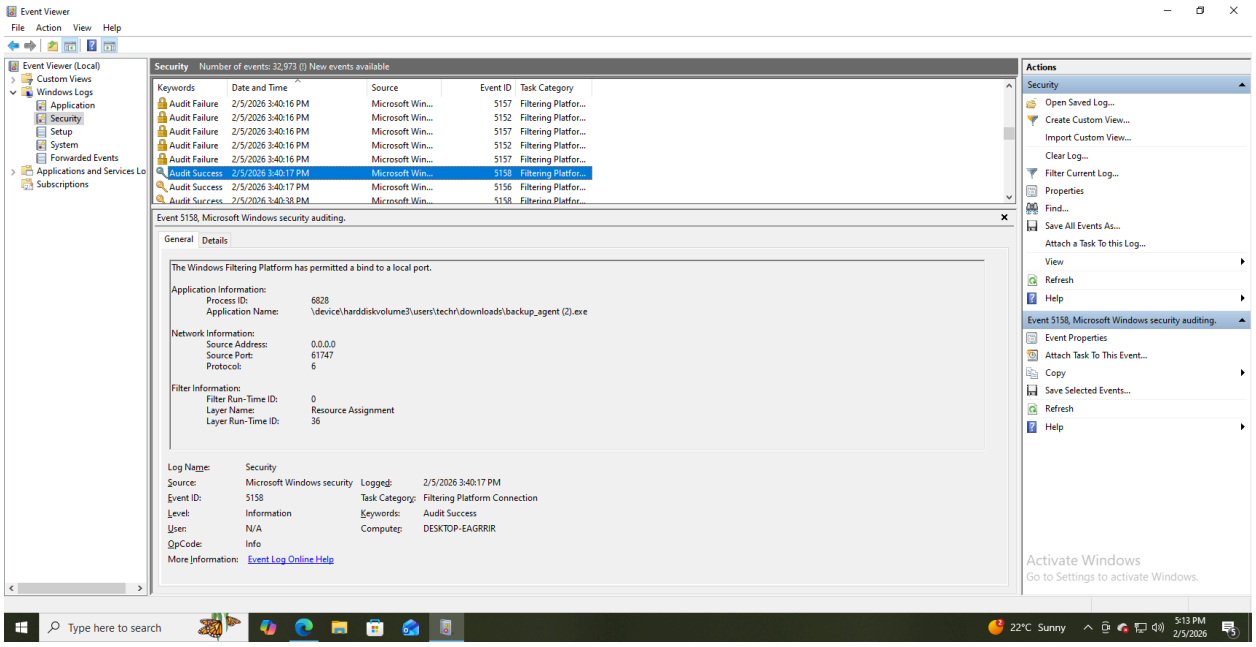
SOC Incident Report



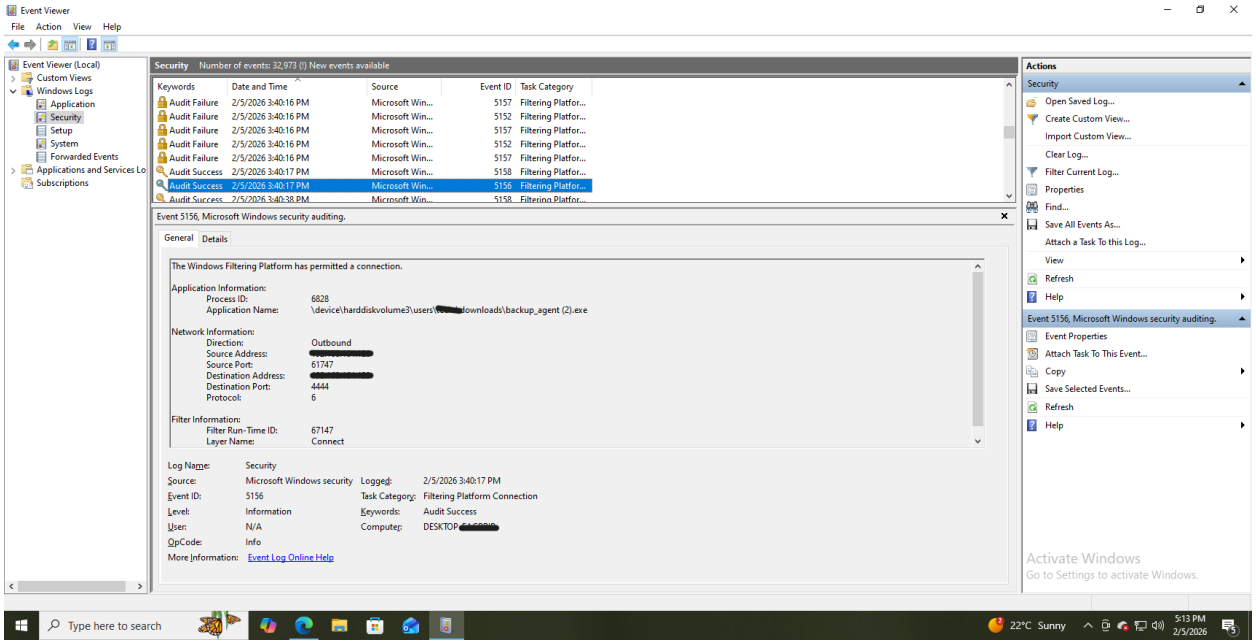


SOC Incident Report

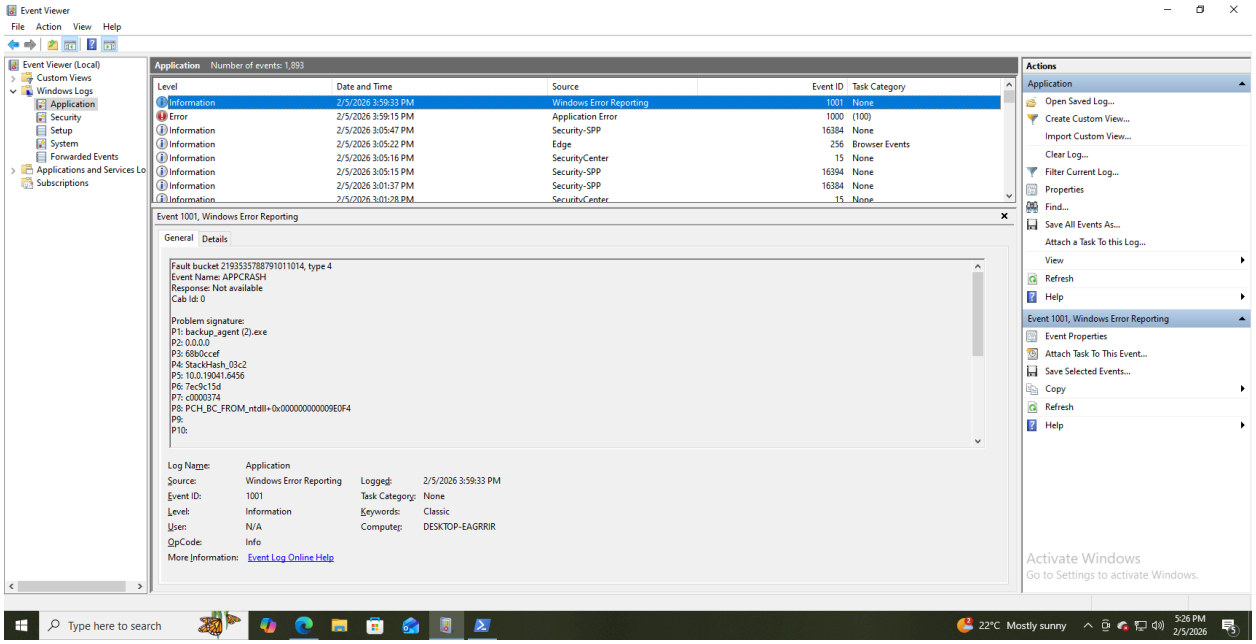


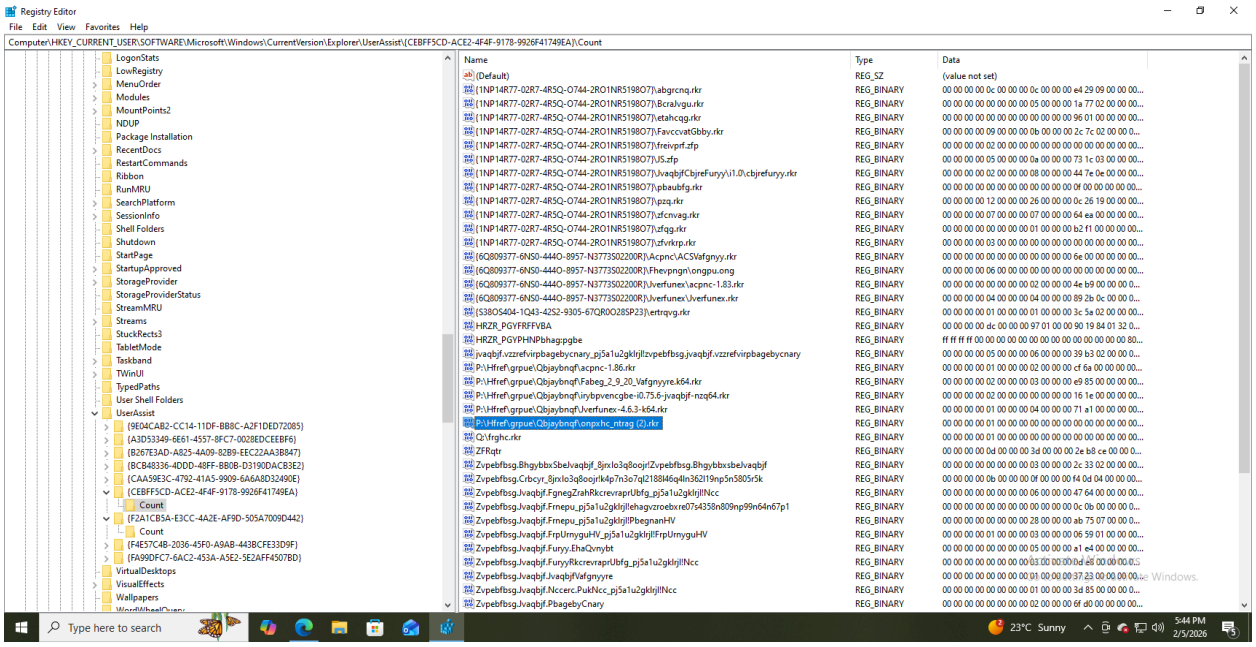


SOC Incident Report



SOC Incident Report





Active Attack Simulation & Forensic Investigation – Detailed Flow (End-to-End)

- Attacker machine (Kali Linux VM) generated a 64-bit Windows reverse TCP payload (`backup_agent.exe`) using msfvenom configured to connect back on port 4444.
- A Python HTTP server was started on Kali (port 80) to host the malicious executable for delivery.
- Windows 10 VM accessed the attacker's HTTP server through a browser and observed open directory listing, confirming exposed file hosting.
- The victim downloaded `backup_agent (2).exe` from the attacker-controlled web server.
- Upon execution of the payload on Windows VM, a reverse TCP connection was initiated from the victim to the attacker on port 4444.
- Metasploit multi/handler on Kali received the incoming connection and successfully established a Meterpreter session.
- Attacker gained remote command execution on Windows 10 (verified using `sysinfo` and `getuid` in Meterpreter).
- A proof-of-compromise file (`hacked.txt`) was created inside `C:\Users\Public\` to demonstrate successful access.
- Additional binary (`discovery.exe` / netcat equivalent) was uploaded to `C:\Users\Public\` for further post-exploitation capability.
- Sensitive system file (`C:\Windows\System32\drivers\etc\hosts`) was downloaded from victim to attacker machine, simulating data exfiltration.
- A screenshot of the compromised Windows system was captured via Meterpreter, demonstrating remote surveillance capability.
- Meterpreter session was terminated after completing post-exploitation activities.
- Wireshark analysis (filtered `tcp.port == 4444`) confirmed TCP 3-way handshake and sustained data exchange between victim ephemeral port (61747) and attacker port 4444.
- Wireshark Conversations tab showed:
    - ~977 packets exchanged
    - ~2 MB data transferred
    - Clear bi-directional communication (A ↔ B traffic)
    - Long session duration (~1157 seconds)
- Windows Defender Firewall logs recorded:
    - Event ID 5158 (bind to local port by backup_agent.exe)
    - Event ID 5156 (permitted outbound connection to attacker IP on port 4444)
- Splunk investigation using:
    - `Process_ID=6828`
    - `Destination_Port=4444`

    - `WinEventLog:Security`
      confirmed outbound malicious connection from backup_agent executable.
- Wazuh dashboard showed active Windows agent and ingested security events for correlation and monitoring.
- Event Viewer (Security log) validated:
    - Filtering Platform permitted connection
    - Application path:
      `\device\harddiskvolume3\users\...\downloads\backup_agent (2).exe`
    - Outbound connection to remote attacker IP on port 4444
- Application log (Event ID 1001 – Windows Error Reporting) showed crash of `backup_agent (2).exe`, indicating instability after execution.
- Registry analysis (UserAssist key):
    - Evidence of execution of `backup_agent (2).exe`
    - Confirms user interaction and program launch from Downloads folder.
- Public folder forensic evidence:
    - Presence of `hacked.txt`
    - Presence of uploaded `discovery.exe`
    - Confirms attacker file creation and persistence artifacts.
- Overall Attack Lifecycle Observed:
    - Payload generation
    - Malicious hosting
    - User download & execution
    - Reverse shell establishment
    - Post-exploitation actions
    - Data exfiltration
    - Log validation & forensic confirmation
