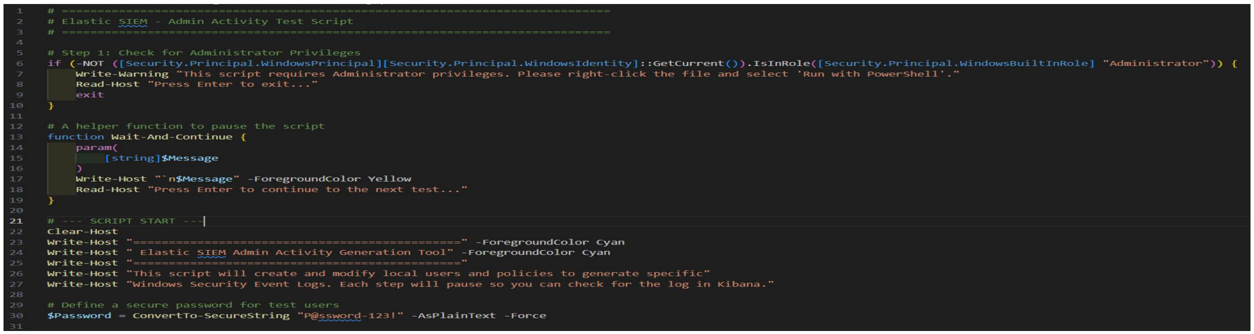
Detection Engineering Use Case Report: Windows Administrative Activities

Ziad Mahmoud Ahmed Abdelgwad

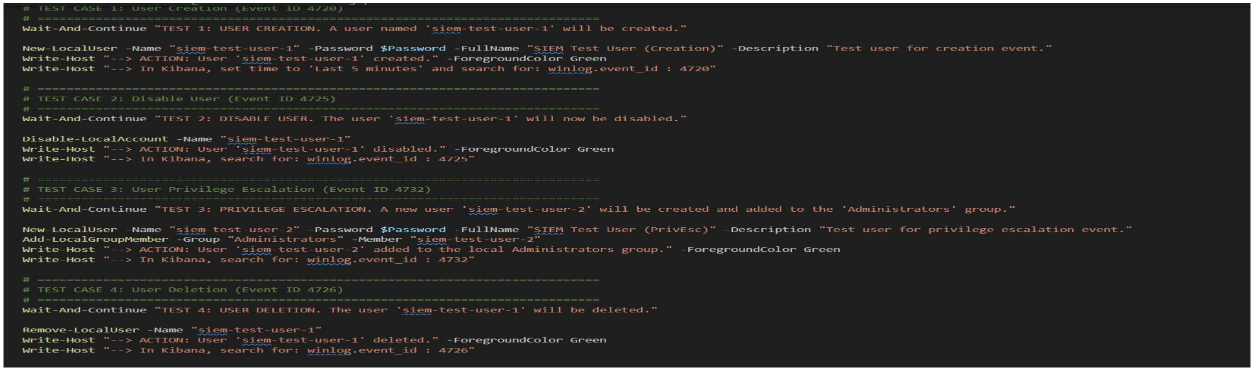
Objective

This document outlines the implementation and verification of seven distinct detection use cases within a newly deployed Elastic SIEM environment. The primary goal is to proactively monitor a Windows host for common administrative and potentially malicious activities. The workflow involves generating specific test events using a custom PowerShell script, Generate-Test-Logs.ps1, creating rules in Kibana to automatically detect this behavior, and verifying the resulting alerts and logs.

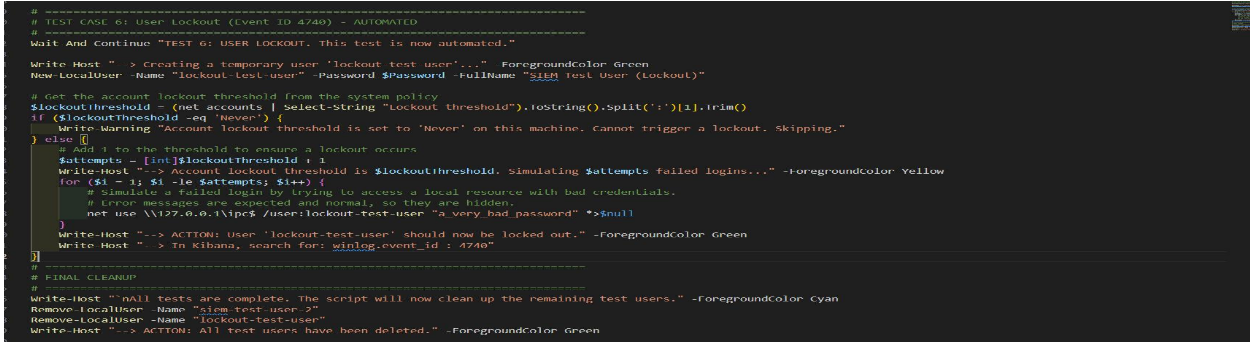
Generate-Test-Logs.ps1:



```
# ===========================================================================
# Elastic SIEM - Admin Activity Test Script
# ===========================================================================

# Step 1: Check for Administrator Privileges
if (-NOT ([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.WindowsBuiltInRole] "Administrator")) {
    Write-Warning "This script requires Administrator privileges. Please right-click the file and select 'Run with PowerShell'."
    Read-Host "Press Enter to exit..."
    exit
}

# A helper function to pause the script
function Wait-And-Continue {
    param(
        [string]$Message
    )
    Write-Host "`n$Message" -ForegroundColor Yellow
    Read-Host "Press Enter to continue to the next test..."
}

# --- SCRIPT START ---
Clear-Host
Write-Host "=============================================" -ForegroundColor Cyan
Write-Host " Elastic SIEM Admin Activity Generation Tool" -ForegroundColor Cyan
Write-Host "============================================="
Write-Host "This script will create and modify local users and policies to generate specific"
Write-Host "Windows Security Event Logs. Each step will pause so you can check for the log in Kibana."

# Define a secure password for test users
$Password = ConvertTo-SecureString "P@ssword-123!" -AsPlainText -Force

```



```
# TEST CASE 1: User Creation (Event ID 4720)
# ===========================================================================
Wait-And-Continue "TEST 1: USER CREATION. A user named 'siem-test-user-1' will be created."

New-LocalUser -Name "siem-test-user-1" -Password $Password -FullName "SIEM Test User (Creation)" -Description "Test user for creation event."
Write-Host "--> ACTION: User 'siem-test-user-1' created." -ForegroundColor Green
Write-Host "--> In Kibana, set time to 'Last 5 minutes' and search for: winlog.event_id : 4720"

# ===========================================================================
# TEST CASE 2: Disable User (Event ID 4725)
# ===========================================================================
Wait-And-Continue "TEST 2: DISABLE USER. The user 'siem-test-user-1' will now be disabled."

Disable-LocalAccount -Name "siem-test-user-1"
Write-Host "--> ACTION: User 'siem-test-user-1' disabled." -ForegroundColor Green
Write-Host "--> In Kibana, search for: winlog.event_id : 4725"

# ===========================================================================
# TEST CASE 3: User Privilege Escalation (Event ID 4732)
# ===========================================================================
Wait-And-Continue "TEST 3: PRIVILEGE ESCALATION. A new user 'siem-test-user-2' will be created and added to the 'Administrators' group."

New-LocalUser -Name "siem-test-user-2" -Password $Password -FullName "SIEM Test User (PrivEsc)" -Description "Test user for privilege escalation event."
Add-LocalGroupMember -Group "Administrators" -Member "siem-test-user-2"
Write-Host "--> ACTION: User 'siem-test-user-2' added to the local Administrators group." -ForegroundColor Green
Write-Host "--> In Kibana, search for: winlog.event_id : 4732"

# ===========================================================================
# TEST CASE 4: User Deletion (Event ID 4726)
# ===========================================================================
Wait-And-Continue "TEST 4: USER DELETION. The user 'siem-test-user-1' will be deleted."

Remove-LocalUser -Name "siem-test-user-1"
Write-Host "--> ACTION: User 'siem-test-user-1' deleted." -ForegroundColor Green
Write-Host "--> In Kibana, search for: winlog.event_id : 4726"
```



```
# ===========================================================================
# TEST CASE 6: User Lockout (Event ID 4740) - AUTOMATED
# ===========================================================================
Wait-And-Continue "TEST 6: USER LOCKOUT. This test is now automated."

Write-Host "--> Creating a temporary user 'lockout-test-user'..." -ForegroundColor Green
New-LocalUser -Name "lockout-test-user" -Password $Password -FullName "SIEM Test User (Lockout)"

# Get the account lockout threshold from the system policy
$lockoutThreshold = (net accounts | Select-String "Lockout threshold").ToString().Split(':')[1].Trim()
if ($lockoutThreshold -eq 'Never') {
    Write-Warning "Account lockout threshold is set to 'Never' on this machine. Cannot trigger a lockout. Skipping."
} else {
    # Add 1 to the threshold to ensure a lockout occurs
    $attempts = [int]$lockoutThreshold + 1
    Write-Host "--> Account lockout threshold is $lockoutThreshold. Simulating $attempts failed logins..." -ForegroundColor Yellow
    for ($i = 1; $i -le $attempts; $i++) {
        # Simulate a failed login by trying to access a local resource with bad credentials.
        # Error messages are expected and normal, so they are hidden.
        net use \\127.0.0.1\ipc$ /user:lockout-test-user "a_very_bad_password" *>$null
    }
    Write-Host "--> ACTION: User 'lockout-test-user' should now be locked out." -ForegroundColor Green
    Write-Host "--> In Kibana, search for: winlog.event_id : 4740"
}
# ===========================================================================
# FINAL CLEANUP
# ===========================================================================
Write-Host "`nAll tests are complete. The script will now clean up the remaining test users." -ForegroundColor Cyan
Remove-LocalUser -Name "siem-test-user-2"
Remove-LocalUser -Name "lockout-test-user"
Write-Host "--> ACTION: All test users have been deleted." -ForegroundColor Green
```

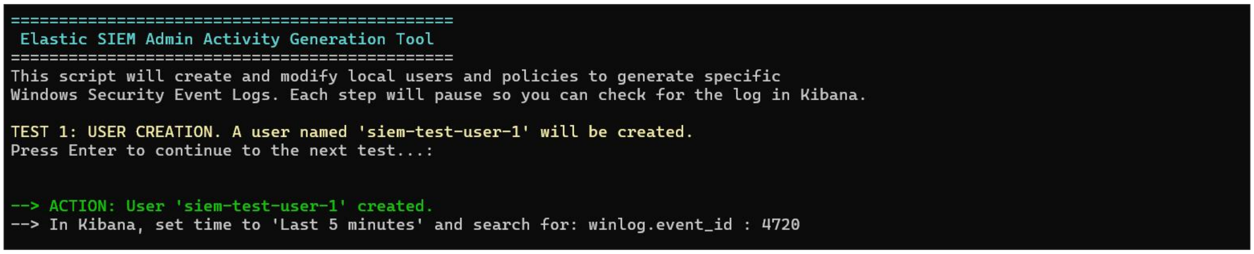
Detection Use Cases :

1. User Account Creation

Rule Name: New User Account Created

Detection Logic (KQL): winlog.event_id : (4720 or 4722)

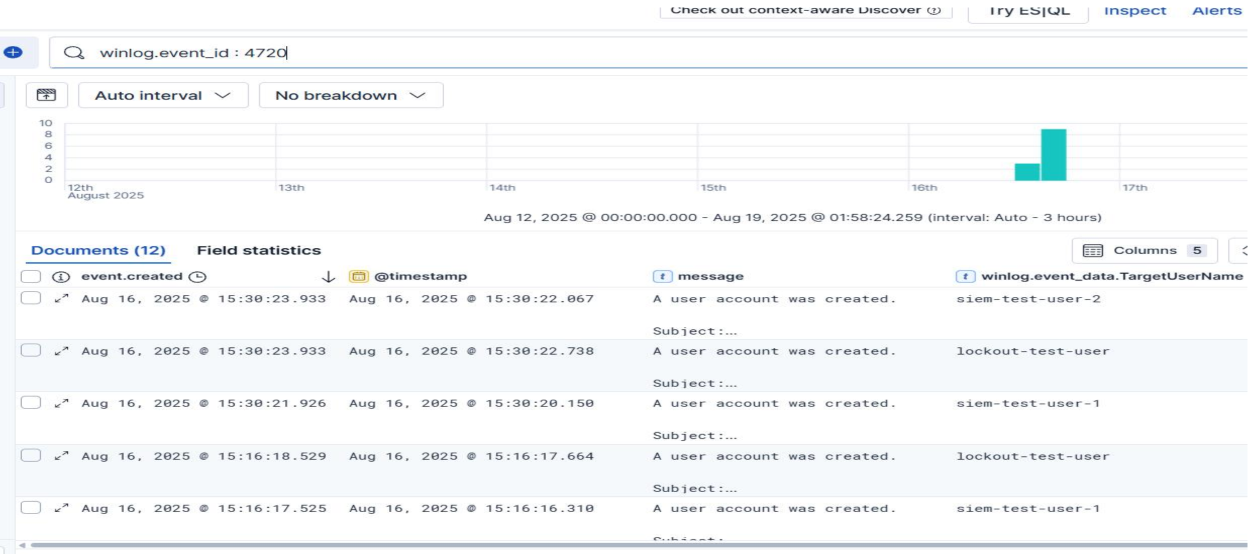
Expected Trigger (Sample Activity): The PowerShell script Generate-Test-Logs.ps1 executes the New-LocalUser cmdlet to create new user accounts such as siem-test-user-1 on the Windows host.



```
=============================================
 Elastic SIEM Admin Activity Generation Tool
=============================================
This script will create and modify local users and policies to generate specific
Windows Security Event Logs. Each step will pause so you can check for the log in Kibana.

TEST 1: USER CREATION. A user named 'siem-test-user-1' will be created.
Press Enter to continue to the next test...:


--> ACTION: User 'siem-test-user-1' created.
--> In Kibana, set time to 'Last 5 minutes' and search for: winlog.event_id : 4720
```

Alert Description: An alert is generated when a Windows Security Event Log with ID 4720 (A user account was created) or 4722 (A user account was enabled) is detected. This activity is crucial for tracking new accounts, which could be a precursor to persistence by an adversary.

Verification: The event was successfully found in Kibana Discover, which returned 12 documents for user creation events, including for users siem-test-user-1, siem-test-user-2, and lockout-test-user.



winlog.event_id : 4720

Aug 12, 2025 @ 00:00:00.000 - Aug 19, 2025 @ 01:58:24.259 (interval: Auto - 3 hours)

Documents (12) Field statistics

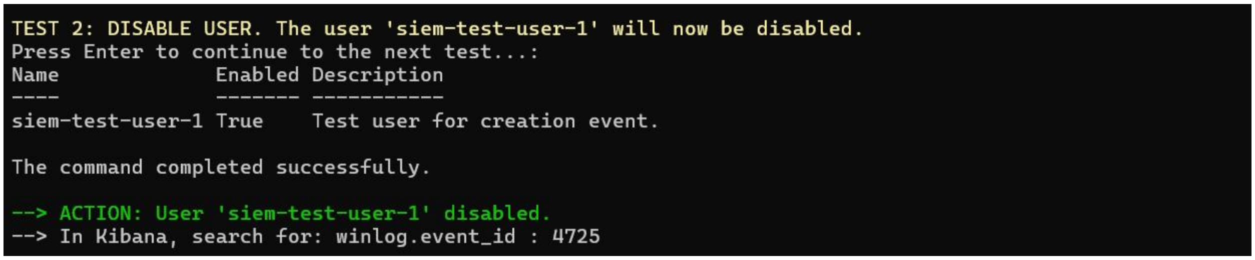
event.created	@timestamp	message	winlog.event_data.TargetUserName
Aug 16, 2025 @ 15:30:23.933	Aug 16, 2025 @ 15:30:22.067	A user account was created. Subject:...	siem-test-user-2
Aug 16, 2025 @ 15:30:23.933	Aug 16, 2025 @ 15:30:22.738	A user account was created. Subject:...	lockout-test-user
Aug 16, 2025 @ 15:30:21.926	Aug 16, 2025 @ 15:30:20.150	A user account was created. Subject:...	siem-test-user-1
Aug 16, 2025 @ 15:16:18.529	Aug 16, 2025 @ 15:16:17.664	A user account was created. Subject:...	lockout-test-user
Aug 16, 2025 @ 15:16:17.525	Aug 16, 2025 @ 15:16:16.310	A user account was created. Subject:...	siem-test-user-1

2. User Account Disabled

Rule Name: User Account Disabled

Detection Logic (KQL): winlog.event_id : 4725

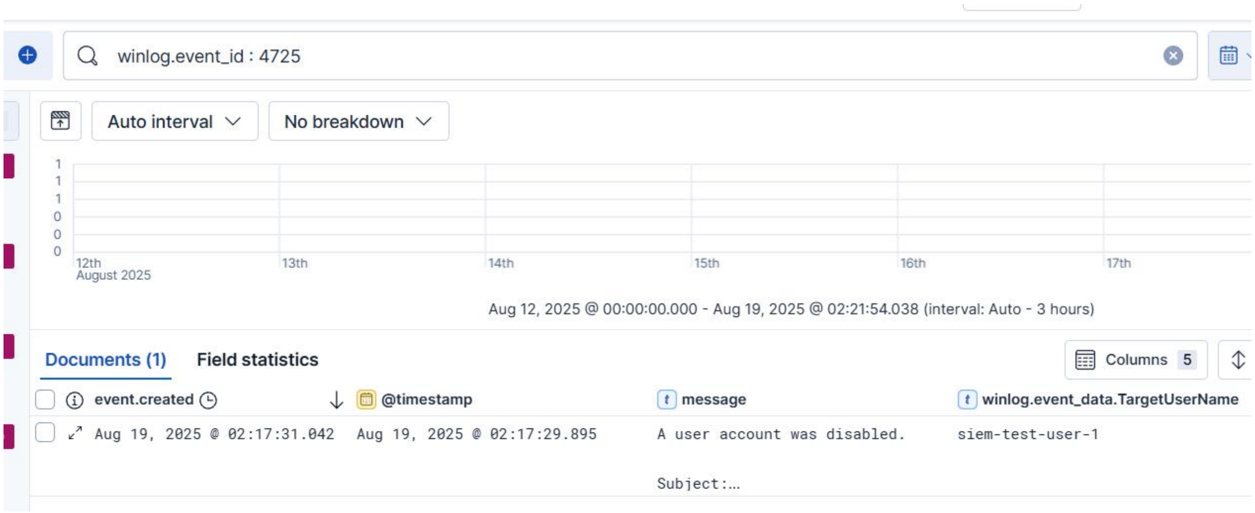
Expected Trigger (Sample Activity): The PowerShell script executes the net user "siem-test-user-1" /active:no command to disable an account.



```
TEST 2: DISABLE USER. The user 'siem-test-user-1' will now be disabled.
Press Enter to continue to the next test...:
Name             Enabled Description
----             ------- -----------
siem-test-user-1 True    Test user for creation event.

The command completed successfully.

--> ACTION: User 'siem-test-user-1' disabled.
--> In Kibana, search for: winlog.event_id : 4725
```

Alert Description: Generates an alert when Windows Event ID 4725 is detected, indicating a user account was disabled. This can be a sign of malicious activity (disrupting user access) or a standard administrative action that should be audited.

Verification: The event was successfully located in Kibana Discover, showing the message "A user account was disabled" for the target user siem-test-user-1.



winlog.event_id : 4725

Aug 12, 2025 @ 00:00:00.000 - Aug 19, 2025 @ 02:21:54.038 (interval: Auto - 3 hours)

Documents (1) Field statistics

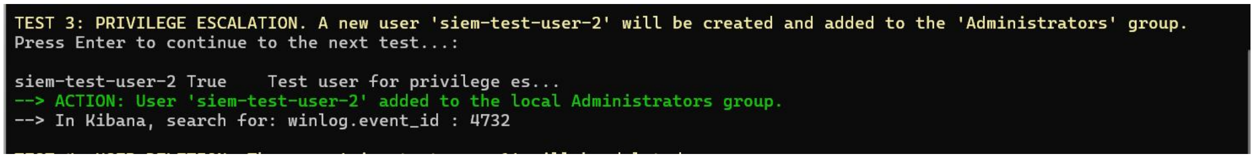
event.created	@timestamp	message	winlog.event_data.TargetUserName
Aug 19, 2025 @ 02:17:31.042	Aug 19, 2025 @ 02:17:29.895	A user account was disabled. Subject:...	siem-test-user-1

3. User Privilege Escalation

Rule Name: Member Added to Local Administrators Group

Detection Logic (KQL): winlog.event_id : 4732

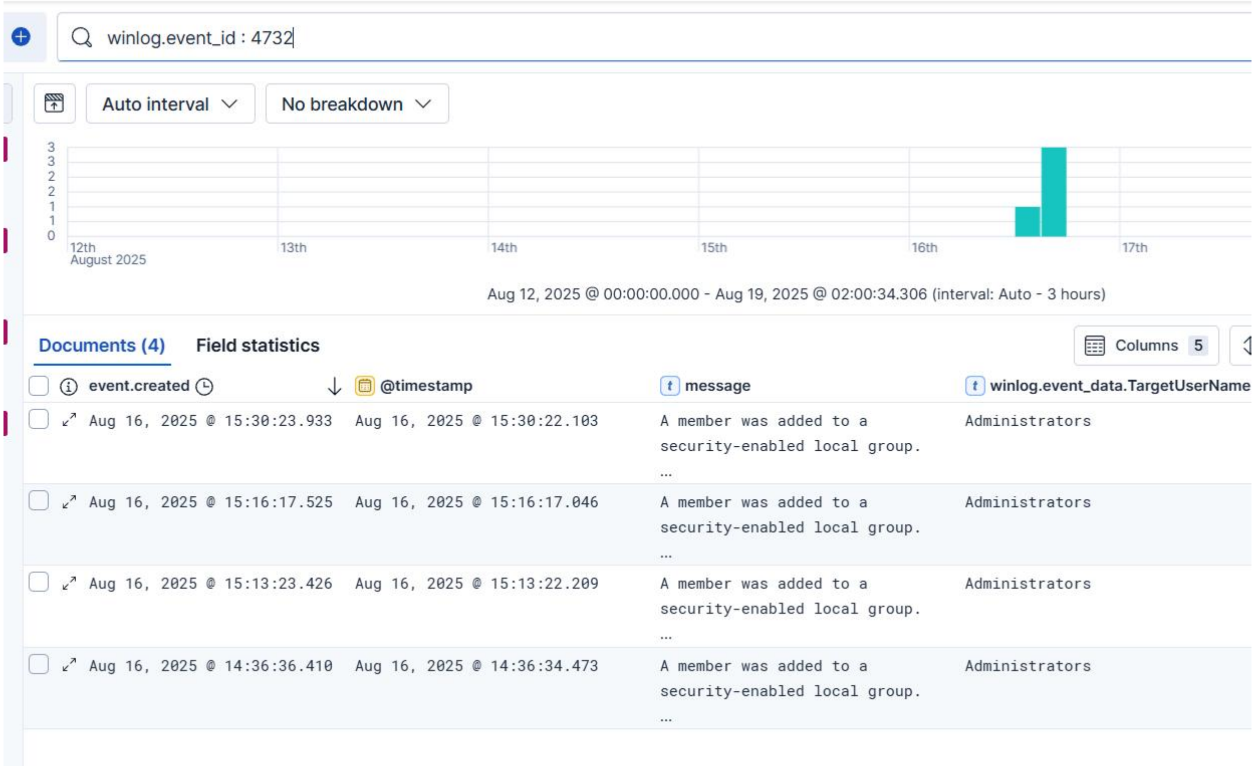
Expected Trigger (Sample Activity): The script creates a new user (siem-test-user-2) and adds it to the local "Administrators" group using the Add-LocalGroupMember cmdlet.



```
TEST 3: PRIVILEGE ESCALATION. A new user 'siem-test-user-2' will be created and added to the 'Administrators' group.
Press Enter to continue to the next test...:

siem-test-user-2 True    Test user for privilege es...
--> ACTION: User 'siem-test-user-2' added to the local Administrators group.
--> In Kibana, search for: winlog.event_id : 4732
```

Alert Description: This alert triggers on Windows Event ID 4732, which indicates a user or group was added to a local security group. Correlating this with the group name "Administrators" signals a critical privilege escalation event.

Verification: The search in Kibana Discover successfully returned 4 events showing a member being added to a security-enabled local group.



winlog.event_id : 4732

Aug 12, 2025 @ 00:00:00.000 - Aug 19, 2025 @ 02:00:34.306 (interval: Auto - 3 hours)

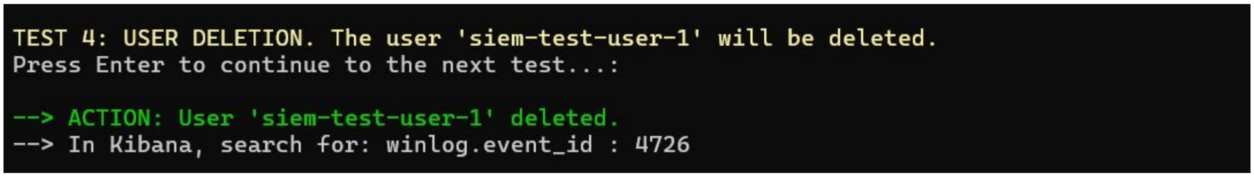
event.created	@timestamp	message	winlog.event_data.TargetUserName
Aug 16, 2025 @ 15:30:23.933	Aug 16, 2025 @ 15:30:22.103	A member was added to a security-enabled local group. …	Administrators
Aug 16, 2025 @ 15:16:17.525	Aug 16, 2025 @ 15:16:17.046	A member was added to a security-enabled local group. …	Administrators
Aug 16, 2025 @ 15:13:23.426	Aug 16, 2025 @ 15:13:22.209	A member was added to a security-enabled local group. …	Administrators
Aug 16, 2025 @ 14:36:36.410	Aug 16, 2025 @ 14:36:34.473	A member was added to a security-enabled local group. …	Administrators

4. User Account Deletion

Rule Name: User Account Deleted

Detection Logic (KQL): winlog.event_id : 4726

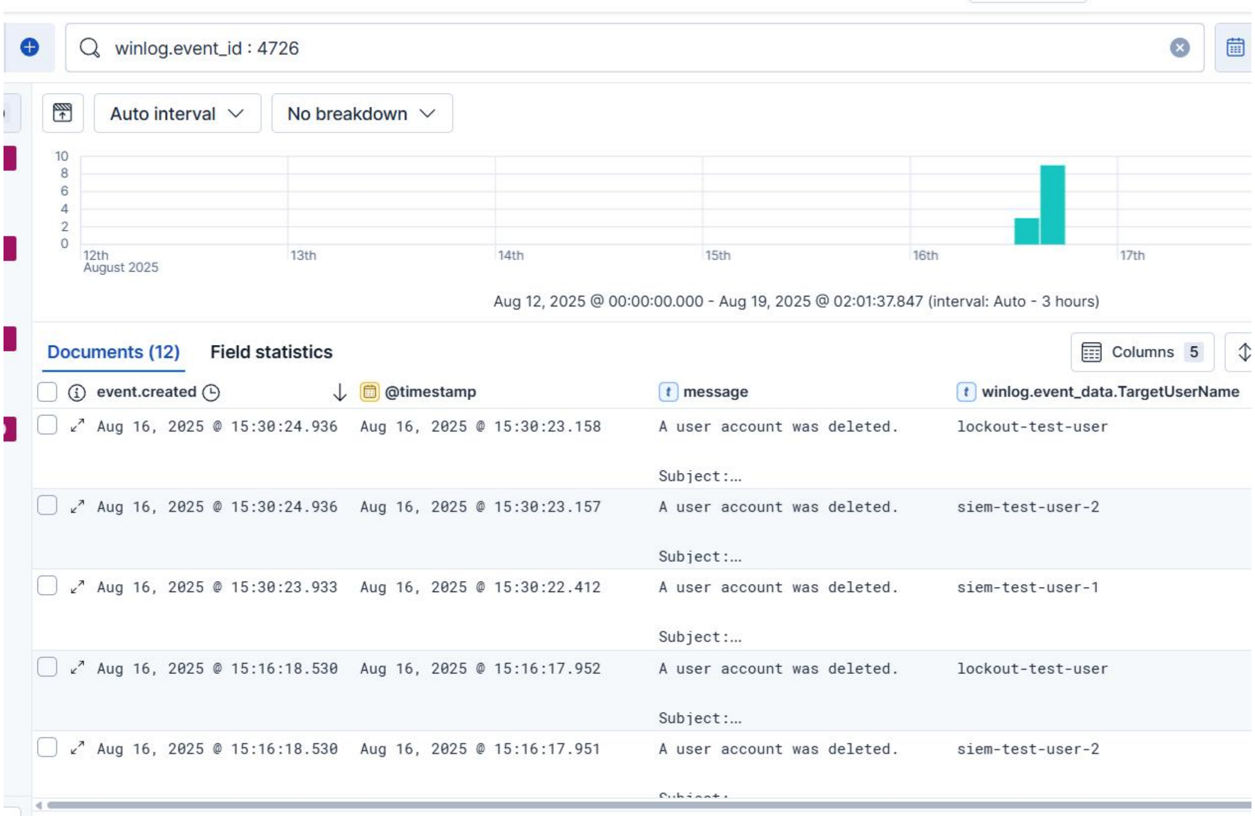
Expected Trigger (Sample Activity): The Remove-LocalUser cmdlet is executed by the script to delete test user accounts.



```
TEST 4: USER DELETION. The user 'siem-test-user-1' will be deleted.
Press Enter to continue to the next test...:

--> ACTION: User 'siem-test-user-1' deleted.
--> In Kibana, search for: winlog.event_id : 4726
```

Alert Description: Triggers on Windows Event ID 4726. The deletion of user accounts should be monitored as it can be part of an attacker's anti-forensic actions or an insider threat.

Verification: The KQL query in Kibana successfully found 12 user deletion events, including for siem-test-user-1, siem-test-user-2, and lockout-test-user.



winlog.event_id : 4726

Aug 12, 2025 @ 00:00:00.000 - Aug 19, 2025 @ 02:01:37.847 (interval: Auto - 3 hours)

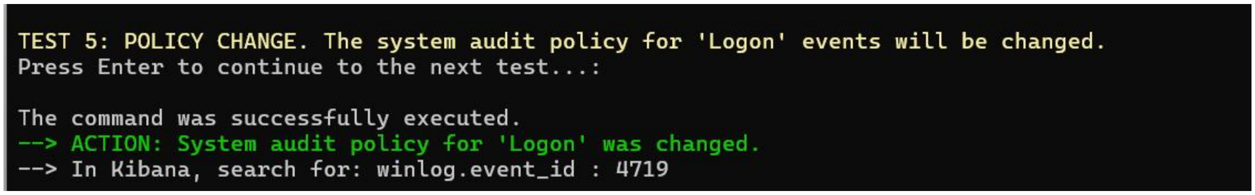
event.created	@timestamp	message	winlog.event_data.TargetUserName
Aug 16, 2025 @ 15:30:24.936	Aug 16, 2025 @ 15:30:23.158	A user account was deleted. Subject:…	lockout-test-user
Aug 16, 2025 @ 15:30:24.936	Aug 16, 2025 @ 15:30:23.157	A user account was deleted. Subject:…	siem-test-user-2
Aug 16, 2025 @ 15:30:23.933	Aug 16, 2025 @ 15:30:22.412	A user account was deleted. Subject:…	siem-test-user-1
Aug 16, 2025 @ 15:16:18.530	Aug 16, 2025 @ 15:16:17.952	A user account was deleted. Subject:…	lockout-test-user
Aug 16, 2025 @ 15:16:18.530	Aug 16, 2025 @ 15:16:17.951	A user account was deleted.	siem-test-user-2

5. Audit Policy Change

Rule Name: System Audit Policy Changed

Detection Logic (KQL): winlog.event_id : 4719

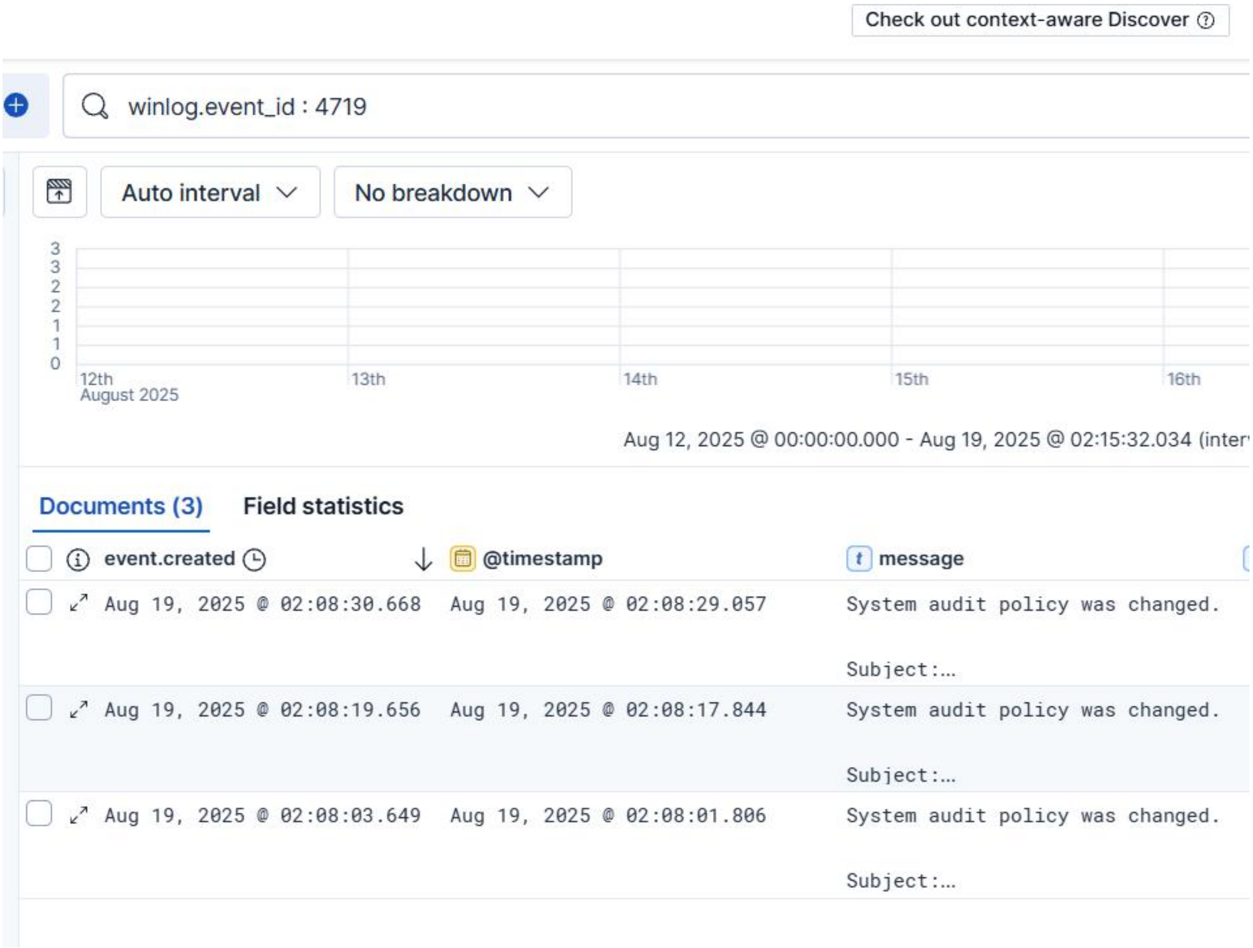
Expected Trigger (Sample Activity): The script executes the native auditpol.exe /set /subcategory:"Logon" ... command to modify the system audit policy.



```
TEST 5: POLICY CHANGE. The system audit policy for 'Logon' events will be changed.
Press Enter to continue to the next test...:

The command was successfully executed.
--> ACTION: System audit policy for 'Logon' was changed.
--> In Kibana, search for: winlog.event_id : 4719
```

Alert Description: This alert triggers on Windows Event ID 4719. Attackers may try to disable security auditing to hide their tracks. Monitoring for changes to the audit policy is a critical security control.

Verification: The search in Kibana successfully returned 3 events confirming that the system audit policy was changed.



winlog.event_id : 4719

Aug 12, 2025 @ 00:00:00.000 - Aug 19, 2025 @ 02:15:32.034 (inter

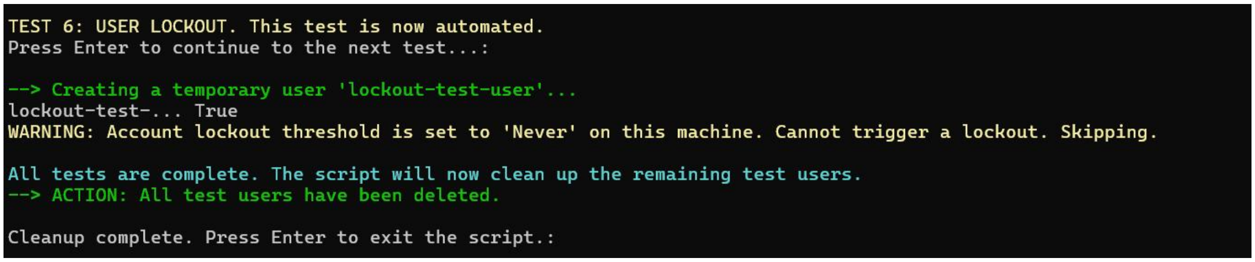
event.created	@timestamp	message
Aug 19, 2025 @ 02:08:30.668	Aug 19, 2025 @ 02:08:29.057	System audit policy was changed. Subject:...
Aug 19, 2025 @ 02:08:19.656	Aug 19, 2025 @ 02:08:17.844	System audit policy was changed. Subject:...
Aug 19, 2025 @ 02:08:03.649	Aug 19, 2025 @ 02:08:01.806	System audit policy was changed. Subject:...

6. User Account Locked Out

Rule Name: User Account Locked Out

Detection Logic (KQL): winlog.event_id : 4740

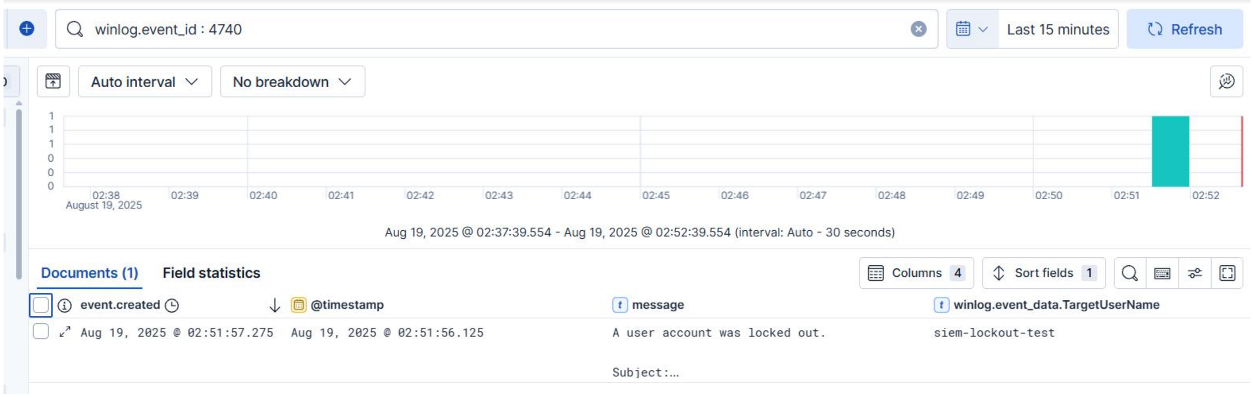
Expected Trigger (Sample Activity): The script creates a test user and then simulates multiple failed login attempts in a loop until the system's account lockout threshold is met, which locks the account.



```
TEST 6: USER LOCKOUT. This test is now automated.
Press Enter to continue to the next test...:

--> Creating a temporary user 'lockout-test-user'...
lockout-test-... True
WARNING: Account lockout threshold is set to 'Never' on this machine. Cannot trigger a lockout. Skipping.

All tests are complete. The script will now clean up the remaining test users.
--> ACTION: All test users have been deleted.

Cleanup complete. Press Enter to exit the script.:
```

Alert Description: Triggers on Windows Event ID 4740. While often caused by user error, a high volume of lockouts for a single user or across multiple users can indicate a password spraying or brute-force attack in progress.

Verification: The search in Kibana successfully found the event showing that the user siem-lockout-test was locked out.



winlog.event_id : 4740

Last 15 minutes

Aug 19, 2025 @ 02:37:39.554 - Aug 19, 2025 @ 02:52:39.554 (interval: Auto - 30 seconds)

event.created	@timestamp	message	winlog.event_data.TargetUserName
Aug 19, 2025 @ 02:51:57.275	Aug 19, 2025 @ 02:51:56.125	A user account was locked out. Subject:...	siem-lockout-test

Bonus Task: Brute Force Correlation Rule

Rule Name: Potential Brute Force Attack Followed by Successful Logon

Detection Logic (EQL):

```
sequence by user.name, source.ip with maxspan=5m
  [authentication where event.outcome == "failure"] with runs=5
  [authentication where event.outcome == "success"]
```

Expected Trigger (Sample Activity): An attacker (or a test script) performs five or more failed login attempts against a single user account from a single source IP address, and then immediately performs a successful login for that same user and IP, all within a five-minute window.

Alert Description: This is a high-fidelity correlation rule that detects a common attack pattern. It identifies a series of password guessing attempts (event.outcome == "failure") that are immediately followed by a successful login (event.outcome == "success"). This significantly reduces false positives compared to just alerting on failed logins and strongly indicates a compromised credential.

Problems, Obstacles, and Solutions Encountered

This project involved significant troubleshooting. The following is a summary of the key obstacles and their solutions.

Obstacle: Logstash failed to start, reporting a 401 Unauthorized error in its logs.

Solution: The password for the elastic user in the Logstash configuration file (02-winlogbeat-input.conf) was incorrect. This was resolved by resetting the elastic user's password and updating the configuration file.

Obstacle: Logstash failed to start, reporting a 400 Bad Request when trying to install an index template.

Solution: The logs revealed a template priority conflict with a pre-existing template in Elasticsearch. This was resolved by adding template_priority => 201 to the elasticsearch output block in the Logstash configuration to ensure the Winlogbeat template took precedence.

Testing (Winlogbeat, PowerShell):

Obstacle: The test script failed because the `Disable-LocalAccount` command was not recognized.

Solution: The modern PowerShell cmdlet was replaced with the more universal legacy command `net user "username" /active:no` in the script.

Obstacle: The automated user lockout test was skipped.

Solution: The script correctly identified that the Windows "Account lockout threshold" was set to 'Never'. The policy was enabled using the command `net accounts /lockoutthreshold:3`.

Kibana Data Visibility:

Obstacle: Logs were not visible in the Kibana Discover tab, which showed "No results match".

Solution: The time range filter was set too narrowly (e.g., "Last 15 minutes"). Expanding the time range to "Today" or "Last 7 days" revealed the logs.

Obstacle: The `event.outcome` field had a `text` data type, causing EQL correlation rules to fail with a mapping conflict.

Solution: The old, incorrectly mapped index (`winlogbeat-v2-*`) was deleted. A new index template was manually created in Elasticsearch to force the `event.outcome` field to be mapped as a `keyword` type for all future indices matching that pattern. This permanently resolved the conflict.