

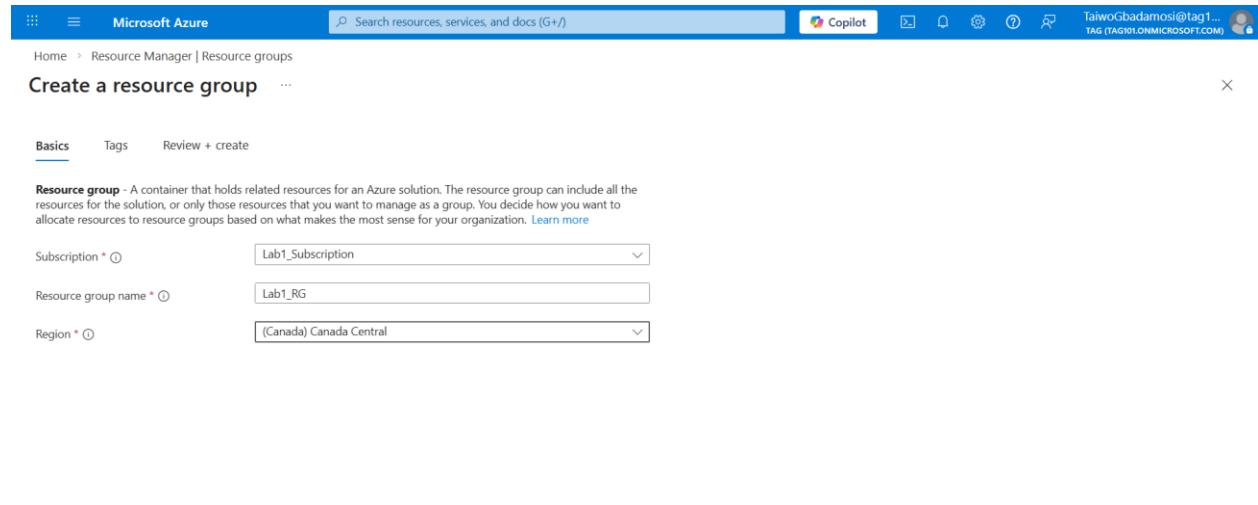
Project Title: Hybrid Cloud SIEM/SOAR Deployment, Log Ingestion & Threat Detection

1. Project Overview & Objectives

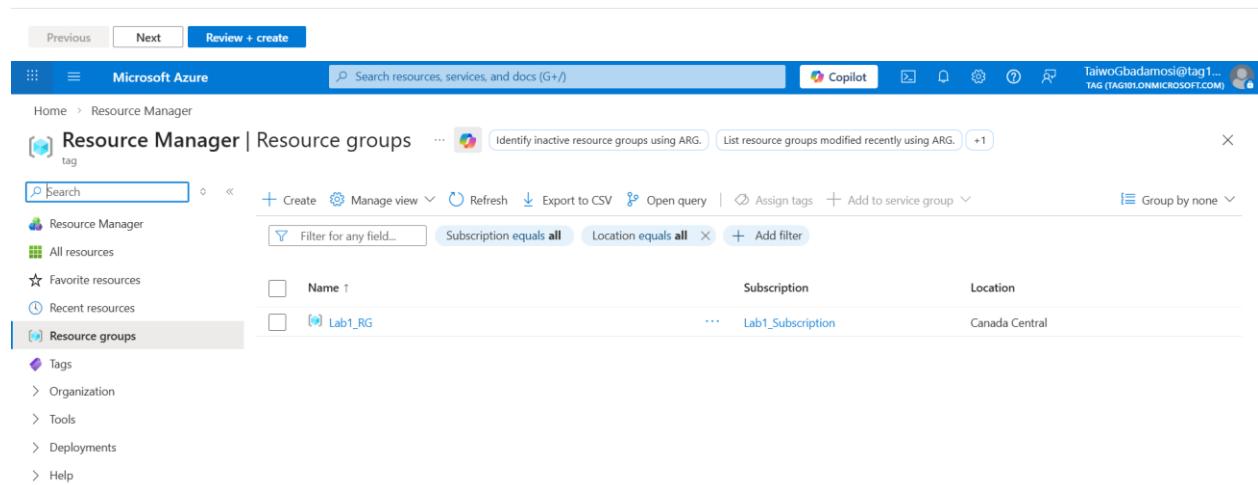
The objective of this lab was to simulate a real-world enterprise security environment. The focus was on centralizing log data from disparate sources, analyzing logs, threat hunting, creating high-fidelity alerts for common attack vectors (Brute Force), and automating the initial steps of the Incident Response lifecycle.

2. Environment Setup & Foundations

Resource Management: Established a centralized Azure Resource Group to manage lifecycle and billing.



The screenshot shows the 'Create a resource group' wizard in the Microsoft Azure portal. The 'Basics' tab is selected. The 'Subscription' dropdown is set to 'Lab1_Subscription'. The 'Resource group name' input field contains 'Lab1_RG'. The 'Region' dropdown is set to '(Canada) Canada Central'. Below the form, there are 'Previous' and 'Next' buttons, and a 'Review + create' button.



The screenshot shows the 'Resource Manager | Resource groups' page in the Microsoft Azure portal. The left sidebar has 'Resource groups' selected. The main area displays a table with one row for the 'Lab1_RG' resource group. The table columns are 'Name' (Lab1_RG), 'Subscription' (Lab1_Subscription), and 'Location' (Canada Central). There are filters at the top of the table: 'Filter for any field...' (with 'Name' selected), 'Subscription equals all' (selected), 'Location equals all' (selected), and '+ Add filter'.

created resource group in Azure for Lab Resources management

Centralized Engine: Provisioned Microsoft Sentinel and Azure Log Analytics Workspace.

The screenshot shows two consecutive steps in the Microsoft Azure portal:

Step 1: Create Log Analytics workspace

This step involves creating a new Log Analytics workspace. The user has selected a subscription ("Lab1_Subscription") and a resource group ("Lab1_RG"). The workspace name is "Lab1-LogAnalyticsWorkspace" and it is being created in the "Canada Central" region.

Step 2: Add Microsoft Sentinel to a workspace

This step involves adding the newly created Log Analytics workspace to an existing Microsoft Sentinel workspace. The user is viewing a list of workspaces, and the "Lab1-LogAnalyticsWorkspace" is selected. A confirmation message at the bottom states: "Deployed Log Analytics Workspace integrated with Microsoft Sentinel for centralized log ingestion and security telemetry analysis".

Retention Policy: Configured workspace settings to balance compliance requirements with cost-effectiveness.

The screenshot shows the Microsoft Azure Log Analytics workspace settings page for 'Lab1-LogAnalyticsWorkspace'. The left sidebar is collapsed, and the main area displays the 'Usage and estimated costs' section. A modal window titled 'Data Retention' is open, showing a slider for 'Data Retention (Days)' set to 30. Below the slider, it says 'Retention for Application Insights data types default to 90 days and will get the workspace retention if it is over 90 days.' There is also a note about setting retention for specific data types. An 'OK' button is at the bottom right of the modal.

The screenshot shows the Microsoft Azure Log Analytics workspace tables settings page for 'Lab1-LogAnalyticsWorkspace'. The left sidebar is collapsed, and the main area displays the 'QualysKnowledgeBase' table settings. Under 'Data retention settings', 'Analytics retention' is set to 'Same as workspace settings (30 days)' and 'Total retention' is set to 'Same as analytics retention (30 days)'. A note says 'No long term retention for this table'. At the bottom are 'Save' and 'Cancel' buttons.

Configured Data Retention policy in Workspace setting for data lifecycle management optimization

3. Threat Intelligence & Governance

Before ingesting logs, I established a Lab test watchlist that could serve as a known-bad baseline in an actual environment

A screenshot of a Microsoft Notepad window titled 'test_watchlist.csv - Notepad'. The menu bar includes File, Edit, Format, View, Help. The main content area contains the following text:

```
Value1
Value2
Value3
Value4
```

Notepad generated Test Watchlist

The screenshot shows the Microsoft Defender interface. On the left, there's a navigation sidebar with sections like Microsoft Sentinel, Threat management, Content management, Configuration, Email & collaboration, and Cloud security. The main area is titled 'My Watchlists' and shows a list of watchlists. One entry is highlighted: 'Lab 1 Test Watchlist' by 'Lab1TestW...' created on '2/15/2026, 4:45:22 PM'. To the right, a detailed view of this watchlist is shown, including its provider (Microsoft), rows (0), and creation time (2/15/2026, 4:45:22 PM). It also includes fields for description, source, created by, last updated, search key, and status.

Test watchlist deployed in Azure

The screenshot shows the Microsoft Azure portal with the Microsoft Sentinel Logs workspace selected. The left sidebar has categories like General, Logs, Threat management, etc. The main area shows a 'New Query 1' pane with a table named 'Tables'. A query is run against it: 'GetWatchlist('Lab1TestWatchlist')'. The results table shows several rows of data, each with a timestamp, ID, and some descriptive values. The results tab is active, and the chart tab is visible above it.

Azure watchlist integrated to Microsoft Sentinel Logs

IoC Management: Manually created a test Threat Indicator object which could ensure the SIEM is primed to detect known malicious fingerprints in an actual environment.

The screenshot shows the Microsoft Defender interface. On the left, there's a navigation sidebar with sections like Threat management, Content management, Configuration, and Automation. The main area shows a search bar and a breadcrumb trail: Search > Watchlists > This page has a new home. A message indicates an upgrade to threat hunting experience. Below this is a 'Filters' section and a 'Indicators (1)' card. The card shows a 'Values' section with 'testloCDomain.com' selected and a 'Name' section with 'TestlocDomain'. To the right, a modal window titled 'Edit indicator' is open. It has a 'Pattern' section with 'Pattern builder' selected. Under 'Domain name', there's a 'Domain name value' field containing 'testloCDomain.com'. Below it is a 'New observable' section with a query: '[domain-namevalue = "testloCDomain.com"]'. There are 'Name' and 'Indicator types' fields, both set to 'TestlocDomain'. At the bottom are 'Save', 'Cancel', and 'Save and duplicate' buttons.

Deploying a test Indicator object

The screenshot shows the Microsoft Azure Sentinel interface. The left sidebar includes sections like General, Threat management, and Logs (which is currently selected). The main area is titled 'Microsoft Sentinel | Logs' and shows a table with the following KQL query results:

```

1 ThreatIntelIndicators
2 | where IsActive == true
3 | where Type == "ThreatIntelIndicators"
4 | project TimeGenerated, IsActive, Pattern, LastUpdateMethod, Data
  
```

The results table shows one row with the following details:

TimeGenerated [UTC]	IsActive	Pattern	LastUpdateM
2026-02-15, 11:18:06.503 p.m.	true	{ domain-namevalue = 'testloC...' }	SentinelRESTA...

At the bottom, there are 'Query details' and '1 - 1 of 1' indicators.

Test Indicator object enriched in Microsoft Sentinel

4. Hybrid Log Ingestion (The "Azure Arc" Implementation)

This phase focused on the Security Reach of the SIEM.

Azure VM: Standard deployment of a Windows VM in Azure.

Create a virtual machine

Validation passed

Basics

Subscription	Lab1_Subscription
Resource group	(new) Lab1_vmg_RG
Virtual machine name	Lab1-vm1
Region	Canada Central
Availability options	No infrastructure redundancy required
Zone options	Self-selected zone
Security type	Standard
Image	Windows 11 Enterprise, version 25H2 - Gen2
VM architecture	x64
Size	Standard D2ds v4 (2 vcpus, 8 GiB memory)
Enable Hibernation	No
Username	localadmin
Public inbound ports	RDP

Estimated monthly costs

Basics	\$91.98
Disks	\$19.71
Networking	\$3.65
Management	\$0.00
Monitoring	\$0.00
Estimated monthly cost	\$115.34

< Previous Next > **Create**

Deployment of Windows 11 vm in Azure

VMware Integration: Deployed a local windows 10 machine and generated an onboarding script via Azure Arc to connect local machine to Azure.

New Virtual Machine Wizard

Ready to Create Virtual Machine
Click Finish to create the virtual machine and start installing Windows 10 x64.

The virtual machine will be created with the following settings:

Name:	Windows 10 x64 (2)
Location:	C:\Users\NEW USER\OneDrive\Documents\Virtual Machine...
Version:	Workstation 17.5 or later
Operating System:	Windows 10 x64
Hard Disk:	60 GB, Split
Memory:	2048 MB
Network Adapter:	NAT
Other Devices:	2 CPU cores, CD/DVD, USB Controller, Sound Card

Customize Hardware...

Power on this virtual machine after creation

< Back **Finish** Cancel

Deployment of Windows 10 vm in vmware

Windows 10 x64 (2) - VMware Workstation

File Edit View VM Tabs Help

Library Type here to search

Home Windows 10 x64 (2)

Administrator: Windows PowerShell

PS C:\Windows\system32> powershell.exe -ExecutionPolicy Bypass -File .\OnboardingScript.ps1

Authentication Complete

Authentication: Directory: C:\Windows

```
Mode LastWriteTime Length Name
---- ----- ----- ----
Directory 'C:\Windows\AzureConnectedMachineAgent' created

Directory: C:\Windows\AzureConnectedMachineAgent

Mode LastWriteTime Length Name
---- ----- ----- ----
Directory 'C:\Windows\AzureConnectedMachineAgent\temp' created
VERBOSER: Installing Azure Connected Machine Agent
VERBOSER: Total Physical Memory: 2048 MB
VERBOSER: .NET Framework version: 4.8.4061
VERBOSER: Checking if this is an Azure virtual machine
VERBOSER: Error: The operation has timed out. checking if we are in Azure
VERBOSER: Creating agent package from https://go.microsoft.com/fwlink/?linkid=213863
VERBOSER: Installing agent package
Installation of agent package succeeded
Connecting machine to Azure... This might take a few minutes.
Cloud: AzureCloud
Verifying connectivity to endpoints that are needed to connect to Azure... This might take a few minutes.
INFO [ ] Please login using the pop-up browser to authenticate.
20N [ ] 1
30N [ ] 1
INFO [ ] Creating resource in Azure...
Correlation ID: Resource ID: /subscriptions/.../resourceGroups/lab1_vm_RG/providers/Microsoft.HybridCompute/machines/DESKTOP-QH550K
```

To direct input to this VM, click inside or press Ctrl+G.

Onboarding Windows vm to azure arc 1

Windows 10 x64 (2) - VMware Workstation

File Edit View VM Tabs Help

Library Type here to search

Home Windows 10 x64 (2)

Administrator: Windows PowerShell

PS C:\Windows\system32> Connect-AzConnectedMachine

Windows PowerShell Copyright (c) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell! <https://aka.ms/powershell>

Microsoft Azure CLI (32-bit) Setup

Completed the Microsoft Azure CLI (32-bit) Setup Wizard

Click the Finish button to exit the Setup Wizard.

Please close and reopen any active terminal window to use Azure CLI.

Back Finish Cancel

To direct input to this VM, click inside or press Ctrl+G.

Onboarding Windows vm to azure arc 2

```

PS C:\Windows\system32> azcmagent show
Resource Name          : DESKTOP-QH1500K
Resource Group Name   : Lab1_vm_RG
Resource Namespace     : Microsoft.HybridCompute
Resource ID            : /subscriptions/00000000-0000-0000-0000-00000000/resourceGroups/Lab1_vm_RG/
Subscription ID        :
Vm ID                 :
Vm Name               :
Correlation ID        :
Vm Type               :
Location              : canadacentral
Cloud                  : AzureCloud
Agent Version         :
Agent Logfile         : C:\ProgramData\AzureConnectedMachineAgent\Log\hinds.log
Agent Status           : Connected
Agent Last Heartbeat  : 2016-02-15T19:06:27-08:00
Agent Error Code       :
Agent Error Details   :
Agent Last Timestamp  :
Using HTTPS Proxy     :
Proxy Bypass List     :
Integrate with SSM    :
Gateway URL           :
Cloud Service          :
Cloud Metadata         :
Manufacturer           : VMware, Inc.
Model                 : VMware ESXi 6.5
MSSQL Server Detected : False
MySQL Server Detected  : False
PGSQL Server Detected : False
Dependent Service Status:
  Agent Service (hinds) : running
  Agent Service (ccmproxy) : running
  GC Service (gcmservice) : running
  SQL Server Host (ccmssql) : down
Portal Page             : https://portal.azure.com/<REDACTED>/resourceGroups/Lab1_vm_RG/providers/Microsoft.HybridCompute/machines/DESKT
Portal Path             : /resourceGroups/Lab1_vm_RG/providers/Microsoft.HybridCompute/machines/DESKT
Disabled Features      :
Agent Auto Upgrade Task Status : enabled, id:<REDACTED>
PS C:\Windows\system32>

```

'Azcmagent show' generating azure arc onboarded info

AMA & DCR: I utilized the Azure Monitor Agent (AMA) and created a Data Collection Rule (AzWinDCR). This ensured that Windows Security Events were streamed directly to the Sentinel Workspace.

Connector details

Windows Security Events via AMA

Setup Advanced options

Prerequisites

To integrate with Windows Security Events via AMA make sure you have:

- ✓ **Workspace data sources:** read and write permissions.
- ⓘ To collect data from non-Azure VMs, they must have Azure Arc installed and enabled. [Learn more](#)

Table management

Manage the table tiers - analytics and lake and table retention. Select the table to manage the table tier and configure its retention settings.

Table name	Tier	Table type	Analytics retention	Total retention	Workspace
SecurityEvent	Analytics	Sentinel	30 days	30 days	lab1-loganalytics

Configuration

Enable data collection rule

Security Events logs are collected only from **Windows** agents.

Refresh

Rule name: AzWinDCR, Created by: Sentinel, Filter name: AllEvents

Create data collection rule

The connector is now connected. Next, review the advanced options to ensure better coverage.

Configure advanced options

Installing and configuring Windows Security Events data collector via Azure Monitoring Agent

Configuring data collection rule for specific resource group (containing both Azure and non-Azure machines)

Creating Data collection rule "AzWinDCR" to ingest specific Windows security event logs

5. Detection & KQL Validation

To ensure logs were flowing correctly, I executed heartbeats and sample queries.

Verification Query: Heartbeat | summarize Count() by Computer

Event Validation: Confirmed that Windows Event ID 4625 (Failed Logons) was populating the SecurityEvent table.

The screenshot shows the Microsoft Sentinel interface. On the left, a sidebar menu is open with 'Logs' selected under 'General'. The main area displays a 'New Query 1' window with the following KQL query:

```
1 Heartbeat
2 | summarize count() by Computer
```

The results pane shows a table with two rows of data:

Computer	count_
DESKTOP-QHJ5ODK	142
Lab1-vm1	4

Windows security event logs integrated to Microsoft Sentinel showing heartbeat of both devices

The screenshot shows the Microsoft Sentinel interface. On the left, a sidebar menu is open with 'Logs' selected under 'General'. The main area displays a 'New Query 1' window with the following KQL query:

```
1 SecurityEvent
```

The results pane shows a table with multiple rows of data:

TimeGenerated [UTC]	Account	AccountType	Computer	EventSourceName
2026-02-16, 3:23:20.203 a.m.	NT AUTHORITY\SYSTEM	Machine	Lab1-vm1	Microsoft-Windows
2026-02-16, 3:23:20.203 a.m.	NT AUTHORITY\SYSTEM	Machine	Lab1-vm1	Microsoft-Windows
2026-02-16, 3:19:48.344 a.m.	NT AUTHORITY\SYSTEM	Machine	Lab1-vm1	Microsoft-Windows
2026-02-16, 3:19:48.344 a.m.	NT AUTHORITY\SYSTEM	Machine	Lab1-vm1	Microsoft-Windows
2026-02-16, 3:19:47.799 a.m.	WORKGROUP\Lab1-vm1\$	Machine	Lab1-vm1	Microsoft-Windows
2026-02-16, 3:19:47.798 a.m.	WORKGROUP\Lab1-vm1\$	Machine	Lab1-vm1	Microsoft-Windows

Query 'SecurityEvent' log integrated from Windows machines

6. SOAR: Automation & Response Orchestration

I implemented SOAR (Security Orchestration, Automation, and Response) to reduce Mean Time to Respond (MTTR).

Sentinel SOAR Essentials

Microsoft Provider | **Microsoft Support** | **3.0.7 Version**

Description: Please refer to the following before installing the solution:
• Review the solution [Release Notes](#)

The Microsoft Sentinel SOAR Essentials solution for Microsoft Sentinel contains Playbooks that can help you get started with basic notification and orchestration scenarios for common use cases. These include Playbooks for sending notifications over email and/or collaboration platforms such as MS Teams, Slack, etc.

Workbooks: 3, **Playbooks:** 23
[Learn more about Microsoft Sentinel](#) | [Learn more about Solutions](#)

Content type: 23 Playbook, 3 Workbook
Category: Security - Automation (SOAR)
Pricing: Free

Manage | **Actions** | [View details](#)

Configuring Sentinel SOAR Essentials data connector to utilize SOAR templates

Incident tasks - Microsoft Defender XDR Ransomware Playbook for SecOps

Description: This playbook add Incident Tasks based on Microsoft Defender XDR Ransomware Playbook for SecOps. This playbook will walk the analyst through four stages of responding to a ransomware incident: containment, investigation, eradication and recovery, and prevention. The step-by-step instructions will help you take the required remedial action to protect information and minimize further risks.

Connectors in use: Microsoft Sentinel

Post Deployment:
 1. Add Microsoft Sentinel Responder role to the managed identity.
 2. Assign playbook to the automation rule.

Source name: SentinelSOARessentials | **Version:** 1.0
Supported By: Microsoft Corporation | **Author:** Microsoft

Create playbook

Configured Test Playbook utilizing SOAR template

Automation rules Active playbooks Playbook templates

+ Create Enable Disable Delete Api Connections

Search: Subscription : Lab1_Subscription Add filter

Name	Status	Plan	Trigger kind	Subscription	Resource group	Location	Source name	Tags
Defender_XDR_Ransomware_Playbook_SecOps	Enabled	Consumption	Microsoft Sentinel In...	Lab1_Subscript...	Lab1_RG	Canada Central	SentinelSOARess...	

Centralized repository of active SOAR playbooks for automated incident response

Playbook Development: Utilized the Sentinel SOAR Essentials template to create a Logic App.

Microsoft Azure Search resources, services, and docs (G+)

Home > From Microsoft 365 Defender > Defender_XDR_Ransomware_Playbook_SecOps

Defender_XDR_Ransomware_Playbook_SecOps | Logic app designer

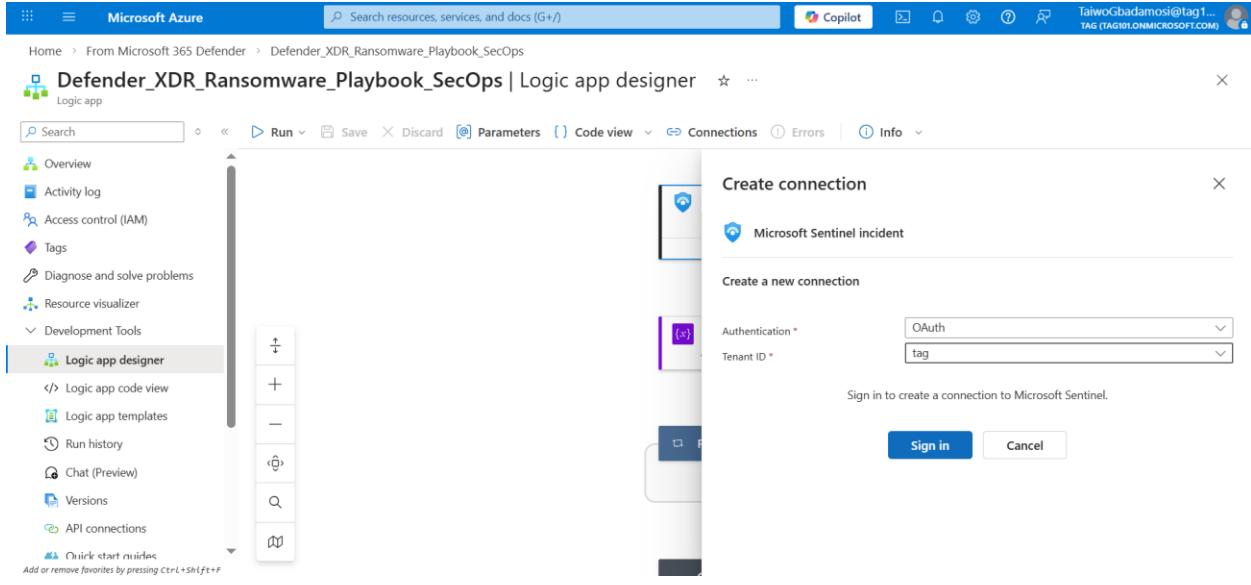
Overview Activity log Access control (IAM) Tags Diagnose and solve problems Resource visualizer Development Tools Logic app designer Logic app code view Logic app templates Run history Chat (Preview) Versions API connections Quick start guides Settings Monitoring Automation Tasks

```

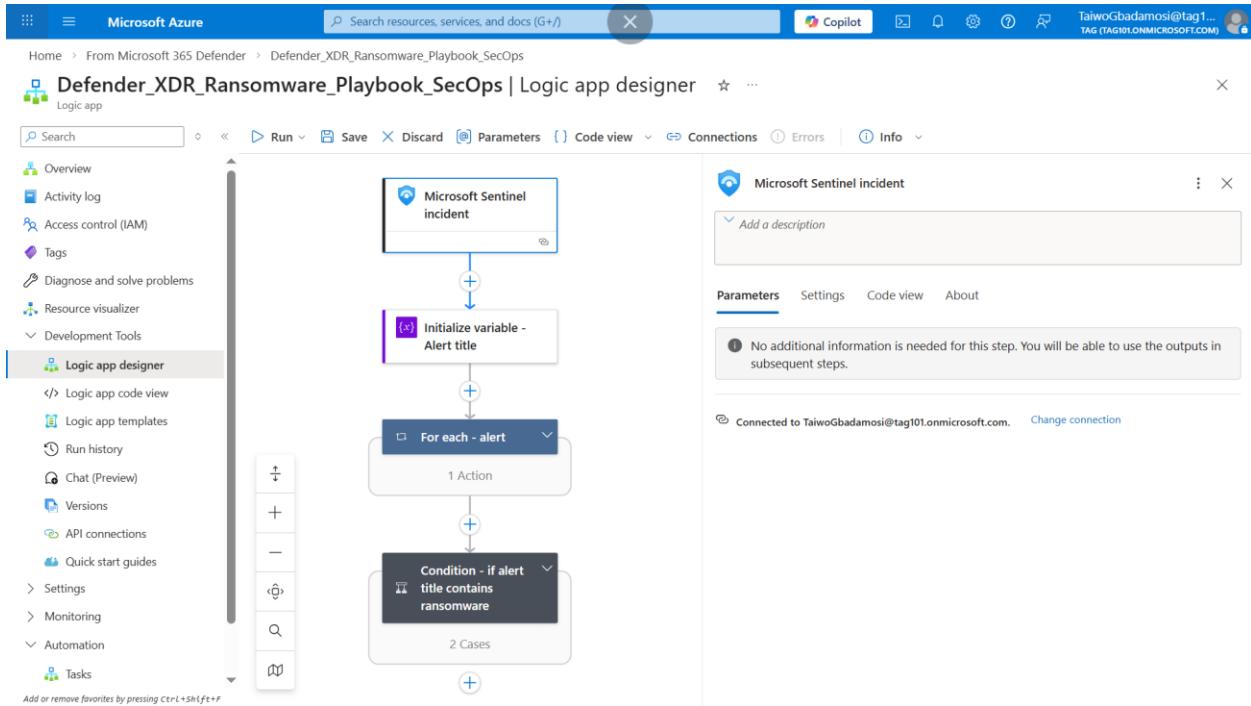
graph TD
    A[Microsoft Sentinel incident] --> B[Initialize variable - Alert title]
    B --> C{For each - alert}
    C --> D[Condition - if alert title contains ransomware]
    D --> E[2 Cases]
  
```

Logic App designer showing configuration of Test playbook Logic flow

Identity Integration: Authenticated the playbook with my Entra ID tenant to allow for identity-based responses.



Established API connections and configured authentication between the SOAR playbook and my Microsoft Entra ID tenant to facilitate automated identity actions.



Playbook authenticated to my Entra id tenant account

Automation Rule: Created a trigger that automatically calls the playbook and assigns specific alerts to the Security Engineer (myself) upon creation.

Configured a Sentinel Automation Rule to trigger the SOAR playbook, enabling an automated response whenever specific alerts are generated.

Configured a Sentinel Automation Rule to automatically assign incidents to myself whenever specific alerts are generated.

Microsoft Defender

Analytics > Analytics > Analytics rule wizard

Validation passed.

Analytics rule wizard - Create a new Scheduled rule

Analytics rule details

- Name: Multiple Unsuccessful Login Detection
- Description: This rule detects multiple unsuccessful login attempts indicating a potential brute force attack attempt
- Severity: Medium
- Status: Enabled

Analytics rule settings

- Rule query:

```
// Define the threshold let FailedThreshold = 5; let TimeRange = 10m; //Timeframe to look for consecutive attempts SecurityEvent | where TimeGenerated > ago(1h) // Filter for Failed Login Attempts | where EventID == 4625 // Focus on the specific machine | where Computer == "Your-VM-Name" | summarize FailedAttempts = count(), FirstFailure = min(TimeGenerated), LastFailure = max(TimeGenerated) by TargetUserName, Computer, ipAddress // Detect 5 or more attempts within the timeframe | where FailedAttempts >= FailedThreshold
```
- Rule frequency: Run query every 5 minutes
- Rule period: Last 10 minutes data
- Rule start time: Automatic
- Rule threshold: Trigger alert if query returns more than 0 results
- Event grouping: Group all events into a single alert
- Suppression: Not configured

Review + create

< Previous Save Cancel

Configuration of Analytics Rule with custom KQL for automated threat detection and incident creation 1

Microsoft Defender

Analytics > Analytics > Analytics rule wizard

Validation passed.

Analytics rule wizard - Create a new Scheduled rule

Rule period

- Last 10 minutes data

Alert details

- Not configured

Incident settings

- Create incidents from this rule: Enabled
- Alert grouping: Tenant default
- Incident correlation: Tenant default

Automated response

- Automation rules:
 - Lab_1 SOAR Automation Response
 - Assigning incidents automation rule

Review + create

< Previous Save Cancel

Configuration of Analytics Rule with custom KQL for automated threat detection and incident creation 2

The screenshot shows the Microsoft Defender Analytics Rules page. At the top, it says "Manage all your rules in one place" and "Rules by severity". Below this is a table of active rules:

Severity	Name	Rule type	Data sources	Tactics	Techniques	Sub techniques	Source name
Low	Excessive Windows Logon Failures	Scheduled	Security Events via Legacy Agent	Credential	T1110		Windows Security
Medium	NRT Security Event	NRT	Security Events via Legacy Agent	Defense Evasion	T1070		Windows Security
Medium	(Preview) Anomaly	ML Behavior	Security Events via AMA	Initial Access			Gallery Content
Medium	(Preview) Anomaly	ML Behavior		Initial Access			Gallery Content

To the right, a specific rule is detailed: "Excessive Windows Logon Failures". It includes a description, data sources (Security Events via Legacy Agent, Windows Security Events via AMA), MITRE ATT&CK (Credential Access), and a rule query:

```
let starttime = ago(8d);
let endtime = ago(1d);
let threshold = 0.333;
let countlimit = 50;
SecurityEvent
| where TimeGenerated >= ago(endtime)
```

Note: You haven't used this template yet; You can use it to create analytics rules.

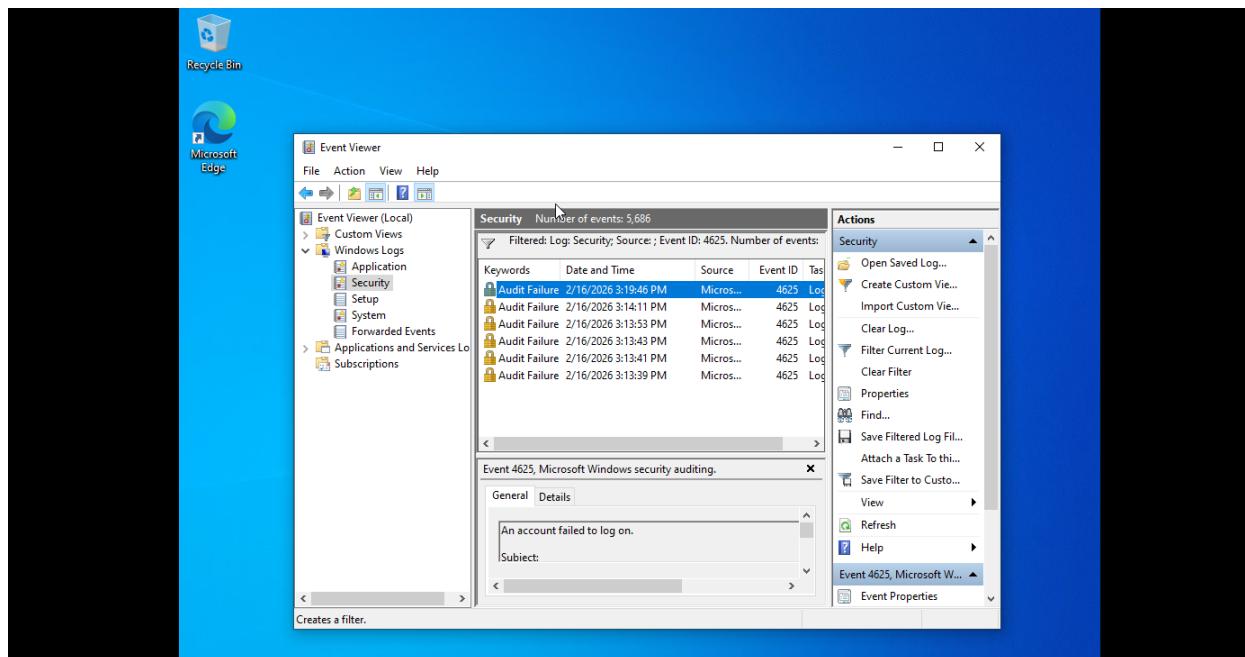
Create rule

Comprehensive view of deployed Analytics Rules and detection logic

7. Attack Simulation & Incident Handling

To test the End-to-End flow, I simulated a Brute Force Attack.

The Attack: Conducted multiple failed login attempts on the VMware-hosted Windows 10 machine.



Local Windows Event Viewer showing Event ID 4625 (Audit Failure) during the simulated attack

The screenshot shows the Microsoft Sentinel Logs interface. On the left, there's a navigation sidebar with sections like General, Logs (which is selected), Guides, Search, Threat management, Incidents, Workbooks, Hunting, Notebooks, Entity behavior, Threat intelligence, and MITRE ATT&CK (Preview). The main area has tabs for Tables and Results. A search bar at the top right says "Time range: Last 3 days" and "Show: 1000 results". Below it is a KQL mode dropdown. The results table shows a list of SecurityEvent logs with columns for TimeGenerated, EventID, Activity, and IPAddress. The first few rows are as follows:

	TimeGenerated [UTC]	EventID	Activity	IPAddress
1	> 2026-02-17, 12:23:06.056 a.m.	4625	4625 - An account failed to log ...	127.0.0.1
2	> 2026-02-17, 12:23:03.856 a.m.	4625	4625 - An account failed to log ...	127.0.0.1
3	> 2026-02-17, 12:23:01.549 a.m.	4625	4625 - An account failed to log ...	127.0.0.1
4	> 2026-02-17, 12:22:58.590 a.m.	4625	4625 - An account failed to log ...	127.0.0.1
5	> 2026-02-17, 12:22:35.035 a.m.	4625	4625 - An account failed to log ...	127.0.0.1
6	> 2026-02-17, 12:22:33.100 a.m.	4625	4625 - An account failed to log ...	127.0.0.1

Microsoft Sentinel logs query showing EventID 4625 (unsuccessful logon attempt) after simulated attack

Detection: The Scheduled Analytics Rule identified the pattern of failures.

Alerting: The Scheduled Analytics rule configured to generate an incident upon threat detection in Sentinel generated an incident, triggered the automation rule, and assigned the case to me.

The screenshot shows the Microsoft Defender Alerts interface. The left sidebar includes icons for Home, Defender XDR, Threats, Vulnerabilities, and Compliance. The main area is titled "Alerts" and displays a list of 26 alerts. The columns include Alert name, Tags, Severity (Medium), Investigation state (New), Status (Initial Access), Category, Detection source (Scheduled detection), Product name (Microsoft Sentinel), and Impacted. All alerts listed are for "Multiple Unsuccessful Login Detection" and are marked as Medium severity and New status. The interface also features a search bar, filter options, and a "Customize columns" button.

High-severity alerts generated by the Analytics Rule following a detected brute-force attempt

Microsoft Defender

Alerts

Defender XDR's built-in alert tuning keeps your SOC focused on high-value incidents by prioritizing actionable signals. New built-in rules are auto-enabled, expanding coverage to additional alert types. Review and adjust the newly added rules in Alert Tuning. Learn more

Export 1 Week ▾

Filter set: Add filter

Impacted assets	First activity	Last activity	Policy name	Classification	Determination	Assigned to	Workspace
	Feb 16, 2026 7:25 PM	Feb 16, 2026 8:25 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	Unassigned	lab1-loganalyticsworkspace
	Feb 16, 2026 7:20 PM	Feb 16, 2026 8:20 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 7:15 PM	Feb 16, 2026 8:15 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 7:10 PM	Feb 16, 2026 8:10 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 7:05 PM	Feb 16, 2026 8:05 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 7:00 PM	Feb 16, 2026 8:00 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:55 PM	Feb 16, 2026 7:55 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:50 PM	Feb 16, 2026 7:50 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:45 PM	Feb 16, 2026 7:45 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:40 PM	Feb 16, 2026 7:40 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:35 PM	Feb 16, 2026 7:35 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace
	Feb 16, 2026 6:30 PM	Feb 16, 2026 7:30 PM	Multiple Unsuccessful Login Detection	Not Set	Not Set	TaiwoGbadamosi@tag101.onmicrosoft.com	lab1-loganalyticsworkspace

Alert service settings

Don't show again

High-severity alerts generated by the Analytics Rule following a detected brute-force attempt automatically assigned to me

The Response: I reviewed the logs, validated the source IP, classified the incident as "Security Testing," and resolved the ticket.

Microsoft Defender

Alerts > Multiple Unsuccessful Login Detection

Part of incident: Multiple Unsuccessful Login Detection [View incident page](#)

What happened

This rule detects multiple unsuccessful login attempts indicating a potential brute force attack attempt

ANALYTICS RULE

Analytics rule details

Rule name: Multiple Unsuccessful Login Detection [View rule in Sentinel](#)

Rule description: This rule detects multiple unsuccessful login attempts indicating a potential brute force attack attempt

Related events

Query results

Manage alert

Status: Resolved

Assign to: TaiwoGbadamosi@tag101.onmicrosoft.com

Classification: Informational, expected activity - Security testing

Comment: This is a security testing for a Lab project

Save Cancel

Incident remediation and closure: Classifying the event as 'Security Testing' and updating the status to 'Resolved' following a successful analysis

Microsoft Defender

Alerts

Defender XDR's built-in alert tuning keeps your SOC focused on high-value incidents by prioritizing actionable signals. New built-in rules are auto-enabled, expanding coverage to additional alert types. Review and adjust the newly added rules in Alert Tuning. Learn more

Export 1 Week

Filter set: Add filter

Alert name	Tags	Severity	Investigation state	Status	Category	Detection source	Product name	Im
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
Multiple Unsuccessful Login Detection		Medium	Initial Access	Resolved	Scheduled detection	Microsoft Sentinel		
NRT rule Lab1 for Incident Creation		Medium	Persistence: Initial Access	Resolved	NRT rules	Microsoft Sentinel		

Incident detail page showing the 'Resolved' status and classification comments

Hunting > Hunting > Hunt

Lab 1 Test Hunt - Suspicious powershell activity

Owner: Taiwo Gbad... Status: New Hypothesis: Unknown

Hunt name: Lab 1 Test Hunt - Suspicious powershell activity

Description: This is a test lab hunt for suspicious powershell activity

Content: 1 Queries | 0 Bookmarks

Last update time: 2/16/2026, 7:35 PM | Created time: 2/16/2026, 7:35 PM

Queries Bookmarks Entities

0 Reconnaissance 0 Resource Development 0 Initial Access 1 Execution 0 Persistence

Search queries Add filter

PowerShell downloads

Implementation of a PowerShell-specific hunting hypothesis within the Microsoft Sentinel Hunting blade

8. Lessons Learned & Outcomes

Agent Evolution: Transitioning to AMA and DCR demonstrated the granular control Azure now provides over what logs are ingested, significantly reducing noise and costs.

Azure Arc Utility: I learned that Azure Arc effectively treats on-premise servers as native Azure objects, simplifying security policy application.

Automation Logic: Configuring the SOAR playbook reinforced the importance of API permissions and Entra ID authentication in the response chain.

Project Outcome: This lab reinforced my ability to manage the full Logs lifecycle from generation on a local VM to ingestion, detection, and automated resolution in the cloud.