CYCRAFT

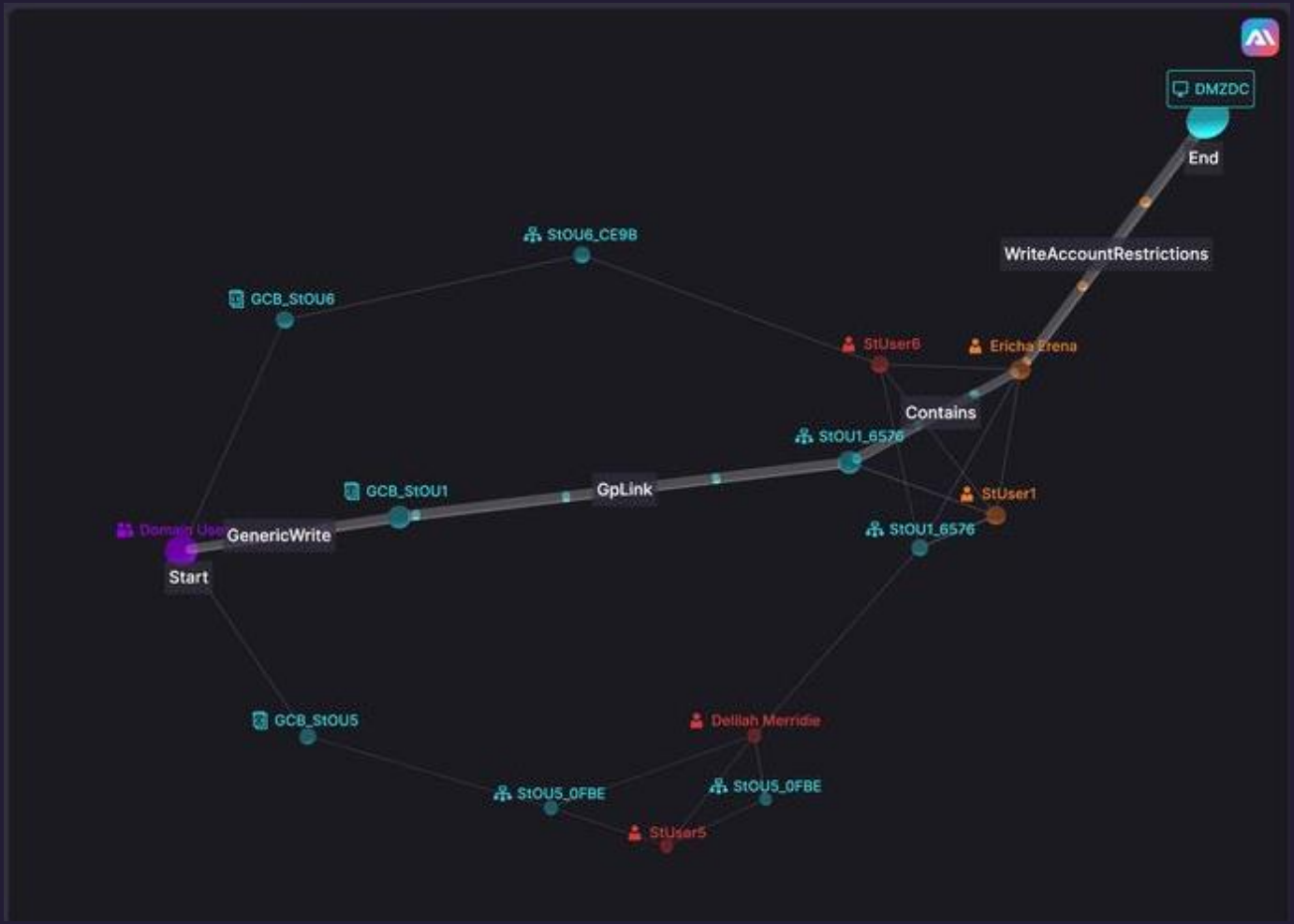
Aztronomy:
Establishing the Foundation of Attack Path Analysis in Azure

June 24th , 2025

Tung-Lin (Echo) Lee

- Lives in Taiwan
- Cyber Security Researcher @ CyCraft
- Focused on Cloud Security
- Speaker at HITCON ENT, ROOTCON
- X: @iflywithoutwind

Attack Path Analysis (APA)



> **Attack Path Analysis**

> Anticipate attacker strategies by mapping potential routes an adversary could use to infiltrate your network systems

CYCRAFT

Edges

> The edge direction represents the flow of **Attack** or **Privilege Escalation**

Entra ID Permission

> Microsoft Entra roles

> Microsoft Graph API Permission

Who has permission to perform which operations

Microsoft Graph API

How to establish a solid foundation for every Edge in Azure Attack Graph

Microsoft Entra Roles

Global Administrator

PRIVILEGED

Actions
microsoft.azure.advancedThreatProtection/allEntities/allTasks
microsoft.azure.informationProtection/allEntities/allTasks
microsoft.azure.serviceHealth/allEntities/allTasks
microsoft.azure.supportTickets/allEntities/allTasks
microsoft.backup/allEntities/allProperties/allTasks
microsoft.cloudPC/allEntities/allProperties/allTasks
microsoft.commerce.billing/allEntities/allProperties/allTasks

- An Entra role is defined by a set of permissions known as **actions**
- An action in Entra ID is a **specific operation** that can be performed on a directory resource
- **No official documentation** describing the mapping between actions and their corresponding APIs

Microsoft Entra built-in roles

Could we assign each action to

custom roles & test abused API

to map actions to specific API calls?

Cannot assign every actions to custom role

```
"microsoft.directory/crossTenantAccessPolicy/partners/create": 201,
"microsoft.directory/privilegedIdentityManagement/allProperties/read": 400,
"microsoft.directory/applications/notes/update": 201,
"microsoft.directory/namedLocations/create": 201,
"microsoft.office365.protectionCenter/attackSimulator/payload/allPropertie
"microsoft.directory/servicePrincipals/oAuth2PermissionGrants/read": 201,
"microsoft.directory/servicePrincipals/owners/update": 201,
"microsoft.directory/users/ownedDevices/read": 201,
"microsoft.directory/crossTenantAccessPolicy/partners/templates/multiTenant
"microsoft.directory/users/authenticationMethods/delete": 201,
"microsoft.directory/applicationPolicies/delete": 201,
"microsoft.directory/groups.security/assignedLabels/update": 400,
"microsoft.directory/applications/owners/read": 201,
"microsoft.directory/servicePrincipals/oAuth2PermissionGrants/limitedRead"
"microsoft.directory/devices/delete": 201,
"microsoft.directory/users/restore": 400,
"microsoft.directory/deviceLocalCredentials/password/read": 201,
"microsoft.directory/contacts/create": 400,
"microsoft.directory/users/allProperties/read": 400,
"microsoft.windows.updatesDeployments/allEntities/allProperties/read": 400
"microsoft.directory/servicePrincipals/managePasswordSingleSignOnCredentia
"microsoft.directory/servicePrincipals/createAsOwner": 201,
"microsoft.directory/deviceTemplates/deviceInstances/read": 201,
"microsoft.directory/users/directReports/read": 201,
"microsoft.directory/users/invalidateAllRefreshTokens": 400,
```

- Entra ID defines **614** actions in built-in roles, but only **100** of them can be assigned to custom roles

```
Response Status Code: 400
Response Payload: {
  "error": {
    "code": "Request_BadRequest",
    "message": "Action 'microsoft.directory/users/allProperties/read'
is not supported for Custom Role creation.",
    "innerError": {
      "date": "2025-05-25T09:17:10",
      "request-id": "ad925e96-3252-44d3-a248-d9aeb6fe6193",
      "client-request-id": "ad925e96-3252-44d3-a248-d9aeb6fe6193"
    }
```

CYCRAFT

MS Graph API Permission

> Microsoft Graph permissions reference

> MS Graph API documentation

AppRoleAssignment.ReadWrite.All

Expand table

Category	Application	Delegated
Identifier	06b708a9-e830-4db3-a914-8e69da51d44f	84bccea3-f856-4a8a-967b-dbe0a3d53a64
DisplayText	Manage app permission grants and app role assignments	Manage app permission grants and app role assignments
Description	Allows the app to manage permission grants for application permissions to any API (including Microsoft Graph) and application assignments for any app, without a signed-in user.	Allows the app to manage permission grants for application permissions to any API (including Microsoft Graph) and application assignments for any app, on behalf of the signed-in user.
AdminConsentRequired	Yes	Yes

Create unifiedRoleAssignment

07/27/2024

Namespace: microsoft.graph

Create a new unifiedRoleAssignment object.

For the directory (Microsoft Entra ID) provider

Expand table

Permission type	Permissions (from least to most privileged)
Delegated (work or school account)	RoleManagement.ReadWrite.Directory
Delegated (personal Microsoft account)	Not supported.
Application	RoleManagement.ReadWrite.Directory

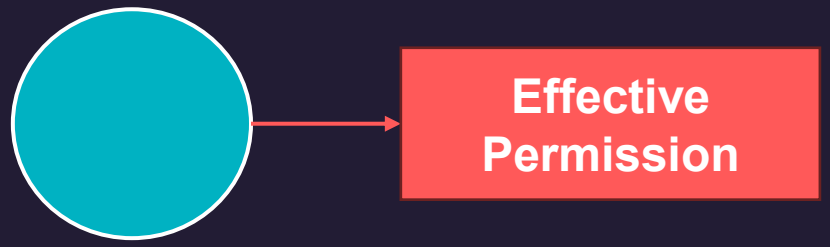
Application & Delegated Permission

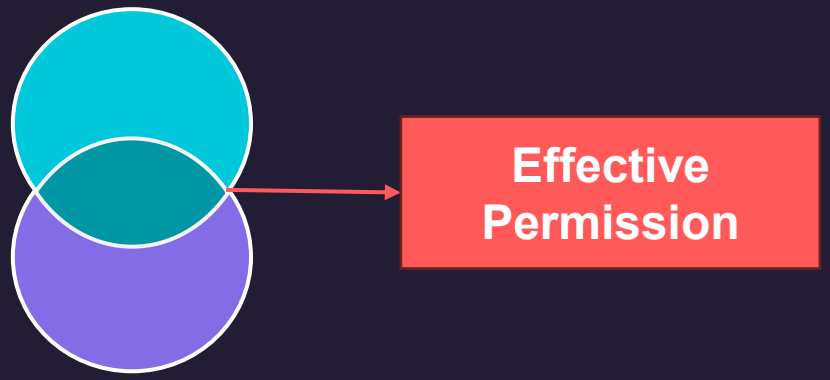
User Permission

(Entra Role)

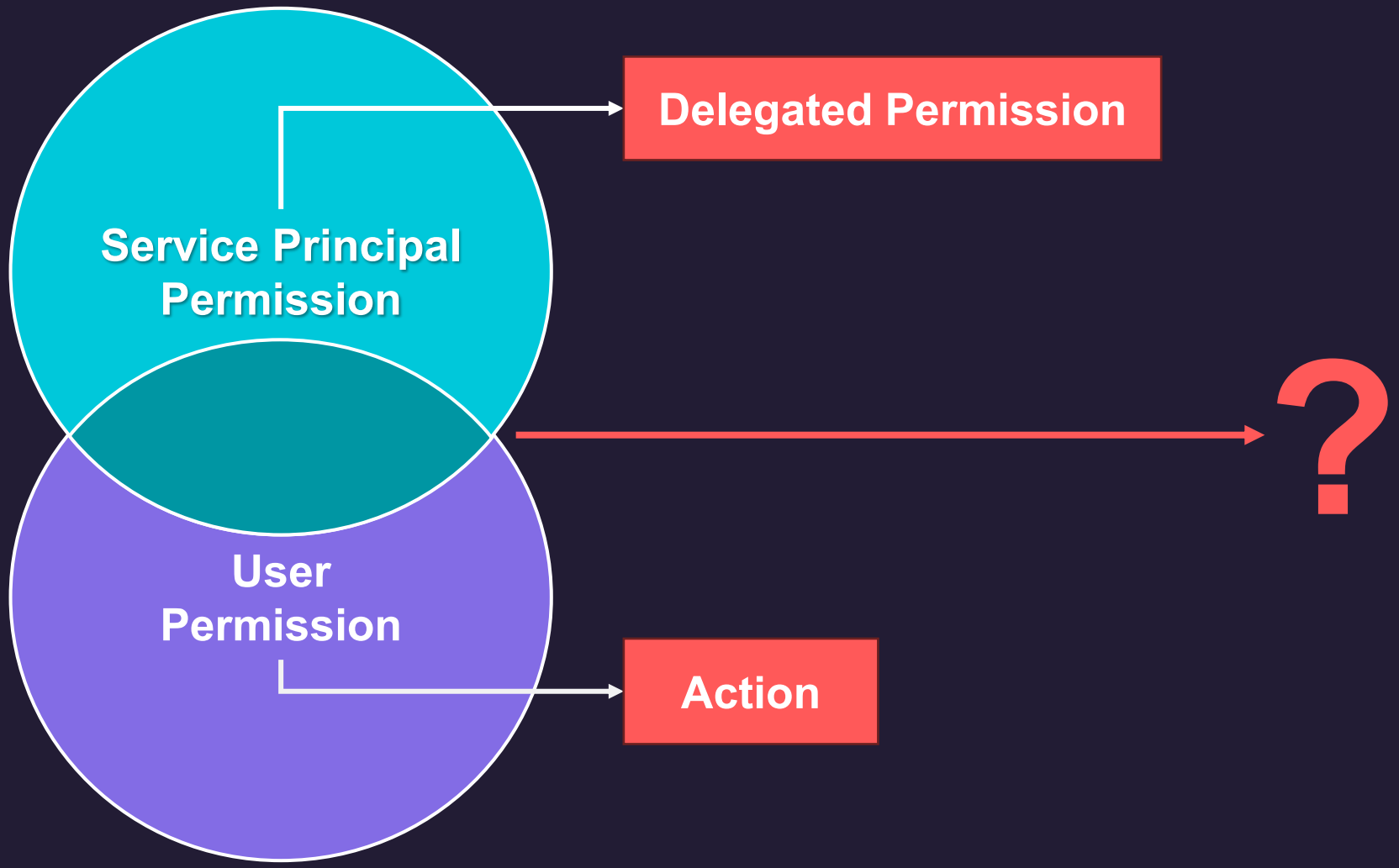
Application Permission

Delegated Permission





Defining the Intersection is Challenging



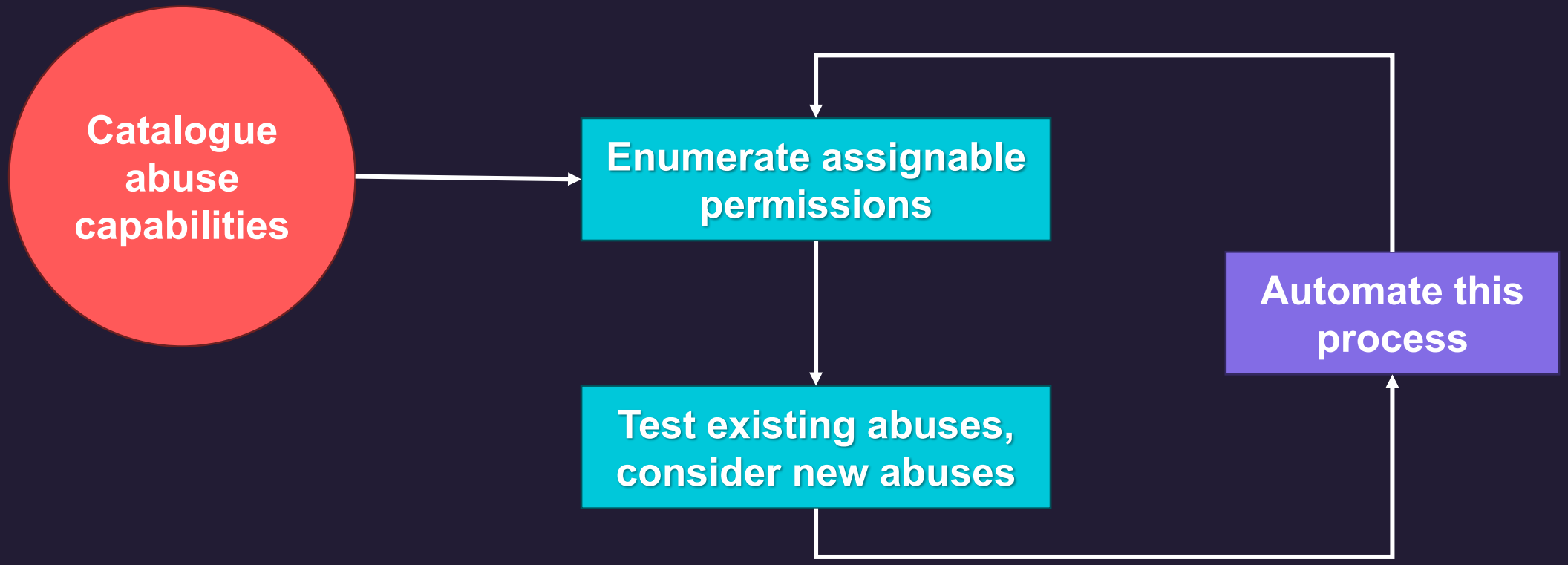
Other Problems We Are Trying To Solve

- The Microsoft documentation may lack clarity or reflect outdated information
  - **Blind Trust of Documentation** Leads to False Positives and Negatives
  - The **evolving dynamics** of Azure IAM systems renders older documentation less reliable
    - Microsoft introduced a bunch of new Graph permissions **(488 → 550)** in February 2025

Continuous Discovery & Validation of Abuse Primitives

BARK (BloodHound Attack Research Kit)

> Process of Building and Running Atomic Tests with BARK:



It's Raining Shells How To Find New Attack Primitives In Azure by Andy Robbins

	A	B	C	D	E	F	G	H	I	J	K
2		Groups				App Registrations		Service Principals		Directory Roles	MS Graph
3		Add Member to Role Eligible Group	Add Owner to Role Eligible Group	Add Member to Non Role Eligible Group	Add Owner to Non Role Eligible Group	Add Owner to App	Add Secret to App	Add Owner to SP	Add Secret to SP	Grant Global Admin Role	Grant MS Graph App Role
4											
5	**AzureAD Admin Roles:**										
6	Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
7	Cloud Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
8	Directory Synchronization Accounts	Failure	Failure	Failure	Failure	Failure	Failure	Success	Success	Failure	Failure
9	Directory Writers	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
10	Global Administrator	Success	Success	Success	Success	Success	Failure	Success	Success	Success	Success
11	Groups Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
12	Identity Governance Administrator	Failure	Failure	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure
13	Hybrid Identity Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Failure	Failure	Failure
14	Intune Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
15	Knowledge Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
16	Knowledge Manager	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
17	Partner Tier1 Support	Failure	Failure	Success	Success	Success	Success	Failure	Failure	Failure	Failure
18	Partner Tier2 Support	Failure	Failure	Success	Success	Success	Failure	Failure	Failure	Success	Success
19	Privileged Role Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Success	Success
20	User Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
21	Windows 365 Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
22											
23	**MS Graph App Roles:**										
24	Application.ReadWrite.All	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
25	AppRoleAssignment.ReadWrite.All	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
26	Directory.ReadWrite.All	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
27	Group.ReadWrite.All	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure
28	GroupMember.Read.All	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
29	GroupMember.ReadWrite.All	Failure	Failure	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure
30	RoleManagement.ReadWrite.Directory	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Success	Success
31	ServicePrincipalEndpoint.ReadWrite.All	Failure	Failure	Failure	Failure	Failure	Failure	Success	Failure	Failure	Failure

The official documentation does not always align with real-world behavior

Types of permissions

Application permissions, also known as app roles, are used in the app-only access scenario, without a signed-in user present. The application is able to access any data that the permission is associated with.

For example, an application granted the Microsoft Graph API's application permission `Files.Read.All` is able to read any file in the tenant using Microsoft Graph. In general, only an administrator or owner of an API's service principal can consent to application permissions exposed by that API.

MS Graph App Roles:	
Application.ReadWrite.All	Failure
AppRoleAssignment.ReadWrite.All	Success
Directory.ReadWrite.All	Failure
Group.ReadWrite.All	Failure
GroupMember.Read.All	Failure
GroupMember.ReadWrite.All	Failure
RoleManagement.ReadWrite.Directory	Success
ServicePrincipalEndpoint.ReadWrite.All	Failure

	MS Graph Grant MS Graph App Role
AzureAD Admin Roles:	
Application Administrator	Failure
Cloud Application Administrator	Failure
Directory Synchronization Accounts	Failure
Directory Writers	Failure
Global Administrator	Success
Groups Administrator	Failure
Identity Governance Administrator	Failure
Hybrid Identity Administrator	Failure
Intune Administrator	Failure
Knowledge Administrator	Failure
Knowledge Manager	Failure
Partner Tier1 Support	Failure
Partner Tier2 Support	Success
Privileged Role Administrator	Success
User Administrator	Failure
Windows 365 Administrator	Failure
MS Graph App Roles:	
Application.ReadWrite.All	Failure
AppRoleAssignment.ReadWrite.All	Success
Directory.ReadWrite.All	Failure
Group.ReadWrite.All	Failure
GroupMember.Read.All	Failure
GroupMember.ReadWrite.All	Failure
RoleManagement.ReadWrite.Directory	Success
ServicePrincipalEndpoint.ReadWrite.All	Failure

It's Raining Shells How To Find New Attack Primitives In Azure by Andy Robbins

CYCRAFT

If you make tooling based on documentation, it will be inaccurate. It'll be wrong, and the people who use your tooling, including yourself, will be extraordinarily frustrated, so

you have to go beyond the documentation

It’s Raining Shells How To Find New Attack Primitives In Azure

Andy Robbins@Insomnihack 2022

Reinvent the wheel

- **Programming Language:**
  - **Build in python**
- **Purpose:**
  - Only Focus on Entra ID
- **Build the testing environment:**
  - Create Target Object
    (Not all of them)
  - Clean up after the tests
- **Output processing:**
  - **Only a 403 Forbidden response is considered a failure;** all other errors will be logged for debugging purposes

Refactor the code to be asynchronous

```
18:19:59 abuse] Abuse_Test_Type: Promote self to GA, Response_Code: 403
18:19:59 entra_id] Finish Abuse Tests of Role: UserShiftPreferences.Read
sed Test Completed 555 / 564 !
18:20:00 abuse] Abuse_Test_Type: Add secret to SP, Response_Code: 403
18:20:00 abuse] Abuse_Test_Type: Promote self to GA, Response_Code: 403
18:20:00 entra_id] Finish Abuse Tests of Role: VirtualAppointment.Read.A
: Completed 557 / 564 !
18:20:00 abuse] Abuse_Test_Type: Promote self to GA, Response_Code: 403
18:20:00 entra_id] Finish Abuse Tests of Role: User.ReadWrite.CrossCloud
Completed 547 / 564 !
18:20:00 abuse] Abuse_Test_Type: Grant self MG App Role, Response_Code:

18:20:00 abuse] Abuse_Test_Type: Promote self to GA, Response_Code: 403
18:20:00 entra_id] Finish Abuse Tests of Role: UserAuthenticationMethod.
Abused Test Completed 550 / 564 !
18:20:00 entra_id] Completed All Role Test !!!
18:20:00 entra_id] Start Generating Output File
18:20:00 entra_id] Output File Name: result\All_ENTRA_ID_ABUSE_TESTS_174
00.json
18:20:00 entra_id] Finish Generating Output File
18:20:00 construct] Start Listing All Omission Resources
18:20:00 construct] Found 4 Group Resources
18:20:01 construct] Found 565 App Resources
18:20:02 construct] No Omission User Found
18:20:02 construct] Finish Listing All Omission Resources
18:20:02 entra_id] Finish Deleting All Experimental Group, Application &
pal
18:20:02 main] Total Abuse Test Time: 105.75322341918945 (s)
```

- **Benefits**
  - **Shorten the experiment duration from 1 hour to about 2 minutes**
- **Drawbacks**
  - Encountered a concurrency violation **(only occurs with sufficient permissions)**
  - Requests may be rate-limited if tested multiple times within a short period

Cloud is ever-changing

> Comparing the test results from **August 2022** to the **latest results**

	Security Gorups				App Registrations		Service Principals		MS Graph	Directory Roles
	Add Member to Non-Role-Assignable Group	Add Owner to Non-Role-Assignable Group	Add Member to Role-assignable Group	Add Owner to Role-assignable Group	Add Owner to App	Add Secret to App	Add Owner to SP	Add Secret to SP	Grant MS Graph App Role	Grant Active Global Admin Role
Azure AD Admin Roles										
Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
Authentication Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Cloud Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
Directory Synchronization Accounts	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Directory Writers	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Exchange Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Global Administrator	Success	Success	Success	Success	Success	Success	Success	Success	Success	Success
Groups Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Hybrid Identity Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Failure	Failure	Failure
Identity Governance Administrator	Failure	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Intune Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Knowledge Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Knowledge Manager	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Partner Tier1 Support	Success	Success	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure
Partner Tier2 Support	Success	Success	Failure	Failure	Success	Success	Failure	Failure	Success	Success
Privileged Role Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Success	Success
SharePoint Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Teams Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
User Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure

Reduced permissions in the Directory Synchronization Accounts role

The Directory Synchronization Accounts role used to have 48 Entra role permissions. It underwent a drastic reduction as a result of the security hardening, with Microsoft removing all of them and replacing them with only one: `microsoft.directory/onPremisesSynchronization/standard/read`. Now that the role only has this read permission, it isn't privileged anymore, which might lead you to assume it is not dangerous.

Directory Synchronization Accounts

Do not use. This role is automatically assigned to the Microsoft Entra Connect service, and is not intended or supported for any other use.

Expand table

Actions	Description
microsoft.directory/onPremisesSynchronization/standard/read	Read and manage objects to enable on-premises directory synchronization

Expand abuse capabilities

from 10 to 30

Grant consent for delegated permissions

- A Microsoft Entra user account with one of the following roles:
  - Privileged Role Administrator, for granting consent for apps requesting any permission, for any API.
  - Cloud Application Administrator or Application Administrator, for granting consent for apps requesting any permission for any API, *except* Microsoft Graph app roles (application permissions).

```python
async def oauth2PermissionGrants(self, session, access_token, principal_id, c
    header = self._generateRequestHeader(access_token)
    body = {
        "clientId": f"{principal_id}",
        "consentType": f"{consent_Type}",
        "resourceId": f"{resource_Id}",
        "scope": f"{scope}",
    }
    async with session.post(
        f'{self._restEndpoint}/oauth2PermissionGrants',
        headers=header,
        json=body
    ) as response:
```

AllPrincipals

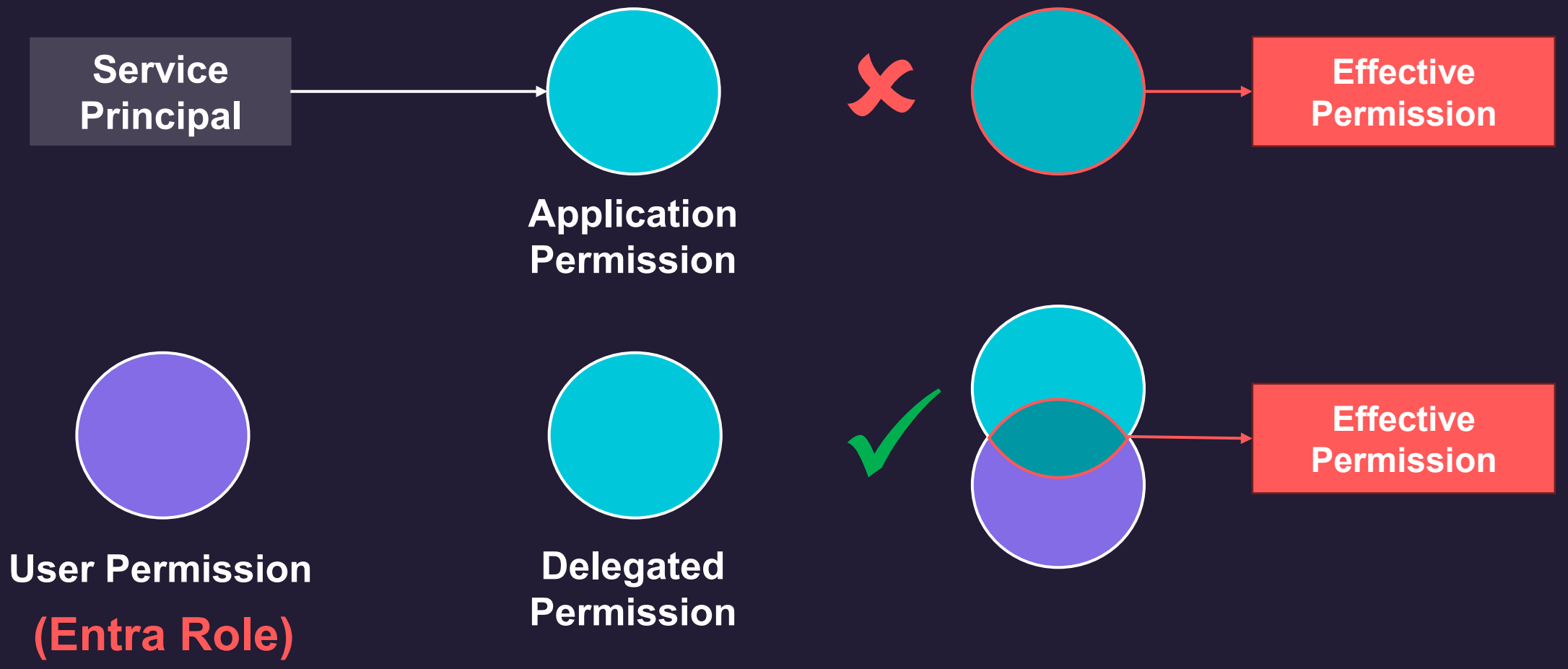
	MS Graph Grant MS Graph Delegated Role
Azure AD Admin Roles	
Application Administrator	Success
Authentication Administrator	Failure
Cloud Application Administrator	Success
Directory Writers	Success
Exchange Administrator	Failure
Global Administrator	Success
Groups Administrator	Failure
Hybrid Identity Administrator	Failure
Identity Governance Administrator	Failure
Intune Administrator	Failure
Knowledge Administrator	Failure
Knowledge Manager	Failure
Partner Tier1 Support	Success
Partner Tier2 Support	Success
Privileged Role Administrator	Success
SharePoint Administrator	Failure
Teams Administrator	Failure
User Administrator	Success
Windows 365 Administrator	Failure

Blind Spot of Abuse Testing via Service

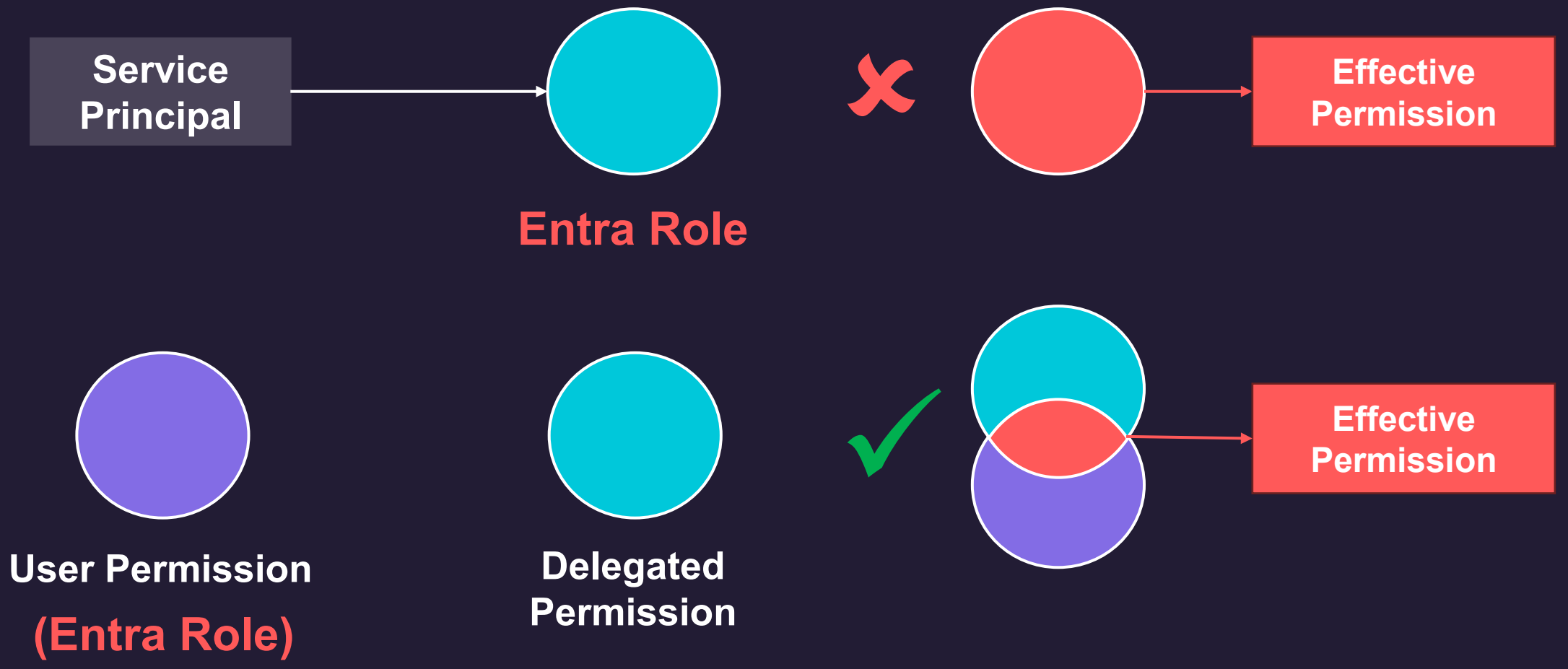
	Devices	Intune		Users
	Retrieve LAPS Password	Execute PS scripts	Assign Intune Role	Reset Password
Azure AD Admin Roles				
Application Administrator	Failure	Failure	Failure	Failure
Authentication Administrator	Failure	Failure	Failure	Failure
Cloud Application Administrator	Failure	Failure	Failure	Failure
Directory Writers	Failure	Failure	Failure	Failure
Global Administrator	Failure	Failure	Failure	Success
Helpdesk Administrator	Failure	Failure	Failure	Failure
Intune Administrator	Failure	Failure	Failure	Failure
Partner Tier1 Support	Failure	Failure	Failure	Failure
Partner Tier2 Support	Failure	Failure	Failure	Success
Password Administrator	Failure	Failure	Failure	Failure
Privileged Authentication Administrator	Failure	Failure	Failure	Failure
User Administrator	Failure	Failure	Failure	Success
Windows 365 Administrator	Failure	Failure	Failure	Failure
Yammer Administrator	Failure	Failure	Failure	Failure

- Bark and Our Project perform abuse tests by creating **Service principals** for each Entra roles
- Assigning the **Global Admin role to the Service Principal** doesn't always grant full capabilities
- However, those operations could be **performed via Azure portal**

Myth of Permissions



Myth of Permissions



Abuse Testing via User identity

> **Plan A (Ideal test conditions):**
  > **120 (Entra ID Role) * 612 (MS Graph Delegated Permissions)** entity for abuse Test

> **Plan B:**
  > Assign all delegated Permission to Microsoft Graph Command Line Tools
  > **Out of the valid Delegation Scope**

```
    [+] oauth2PermissionGrants response: 400 {"
error":{"code":"Request_BadRequest","message":"
Value length '35844' is out of the valid range
of '1' to '8000' for property 'DelegationScope'
.","innerError":{"date":"2025-05-28T06:05:38","
request-id":"db32925d-e584-46ab-9c69-9992659e45
```

Plan C: Workaround

Microsoft Graph App Roles														
Application.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Su	Su	Fa	Fa	Fa	Fail	F	F
AppRoleAssignment.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
DelegatedPermissionGrant.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Su	Fa	Fa	Fail	F	F
DeviceLocalCredential.Read.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Suc	F	F
DeviceManagementConfiguration.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
DeviceManagementRBAC.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	S	F
DeviceManagementScripts.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
Directory.Read.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
Directory.ReadWrite.All	Success	Su	Fa	Su	Su	Fa	Fa	Fa	Su	Fa	Fa	Fail	F	F
Group.ReadWrite.All	Success	Su	Fa	Su	Su	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
GroupMember.ReadWrite.All	Failure	Su	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
RoleEligibilitySchedule.ReadWrite.Directory	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Su	Fail	F	F
RoleEligibilitySchedule.Remove.Directory	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Su	Fail	F	F
RoleManagement.ReadWrite.Directory	Failure	Fa	Su	Fai	Su	Fa	Fa	Fa	Fa	Su	Su	Fail	F	F
ServicePrincipalEndpoint.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Su	Fa	Fa	Fa	Fail	F	F
UserAuthenticationMethod.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User-ConvertToInternal.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User.DeleteRestore.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User.ManageIdentities.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User.Read.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	F
User-PasswordProfile.ReadWrite.All	Failure	Fa	Fa	Fai	Fai	Fa	Fa	Fa	Fa	Fa	Fa	Fail	F	S

- Assign delegated permissions where the equivalent application permission is known to be abusable
- **It's not perfect, but it’s practical**

user_impersonation

in Microsoft Graph ?

Directory.AccessAsUser.All

Directory.AccessAsUser.All

Category	Application	Delegated
Identifier	-	0e263e50-5827-48a4-b97c-d940288653c7
DisplayText	-	Access directory as the signed in user
Description	-	Allows the app to have the same access to information in the directory as the signed-in user.

Application permissions tiering

Tiering of Microsoft Graph application permissions **based on known attack paths**.

Tier definition

Important: suspicious permissions that have not been tested are categorized as Tier-0 for safety and marked with " ⚠️ " until t
researched properly.

Color	Tier	Name	Definition
🔴	0	Family of Global Admins	Permissions with a risk of having a direct or indirect path to Global Admin a takeover.
🟠	1	Family of restricted Graph permissions	Permissions with write access to MS Graph scopes or read access to sensitiv email content), but without a known path to Global Admin.

- Allows the app to have the same access to information in the directory as the signed-in user
- Absent from multiple Tier 0 and critical asset lists in Azure, such as **Azure-Tiering**

Azure-Tiering

CYCRAFT

Explosion Radius of Directory.AccessAsUser.All

> With the delegated permission **Directory.AccessAsUser.All**, a user may have **more permissions** than a service principal assigned the same role

	Security Gorups				App Registrations		Service Principals		MS Graph	Directory Roles
	Add Member to Non-Role-Assignable Group	Add Owner to Non-Role-Assignable Group	Add Member to Role-assignable Group	Add Owner to Role-assignable Group	Add Owner to App	Add Secret to App	Add Owner to SP	Add Secret to SP	Grant MS Graph App Role	Grant Active Global Admin Role
Azure AD Admin Roles										
Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
Authentication Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
Cloud Application Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
Cloud Device Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Directory Writers	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Exchange Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Global Administrator	Success	Success	Success	Success	Success	Success	Success	Success	Success	Success
Groups Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Helpdesk Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
Hybrid Identity Administrator	Failure	Failure	Failure	Failure	Success	Success	Success	Success	Failure	Failure
Identity Governance Administrator	Failure	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Intune Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Knowledge Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Knowledge Manager	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Partner Tier1 Support	Success	Success	Failure	Failure	Success	Success	Failure	Failure	Failure	Success
Partner Tier2 Support	Success	Success	Failure	Failure	Success	Success	Failure	Failure	Success	Success
Privileged Authentication Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
Password Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
Privileged Role Administrator	Failure	Failure	Success	Success	Failure	Failure	Failure	Failure	Success	Failure
SharePoint Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Teams Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
User Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Success
Windows 365 Administrator	Success	Success	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure
Yammer Administrator	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure	Failure

CYCRAFT

No Silver Bullet

> For specific resources (e.g., LAPS or Intune), certain abuse tests still require **specific permissions**

Directory. AccessAsUser.All	Devices	Intune	
	Retrieve LAPS Password	Execute PS scripts	Assign Intune Role
Azure AD Admin Roles			
Cloud Application Administrator	Failure	Failure	Failure
Global Administrator	Failure	Failure	Failure
Intune Administrator	Failure	Failure	Failure

Application Permission	Devices	Intune	
	Retrieve LAPS Password	Execute PS scripts	Assign Intune Role
Microsoft Graph App Roles			
DeviceLocalCredential.Read.All	Success	Failure	Failure
DeviceManagementConfiguration.ReadWrite.All	Failure	Success	Failure
DeviceManagementRBAC.ReadWrite.All	Failure	Failure	Success
DeviceManagementScripts.ReadWrite.All	Failure	Success	Failure

Directory. AccessAsUser.All	Devices	Intune	
	Retrieve LAPS Password	Execute PS scripts	Assign Intune Role
Azure AD Admin Roles			
Cloud Application Administrator	Success	Failure	Failure
Global Administrator	Success	Success	Success
Intune Administrator	Success	Success	Success

Assign equivalent delegated permissions

Enable / Disable Security Defaults

Permission type	Higher privileged permissions
Delegated (work or school account)	Policy.ReadWrite.SecurityDefaults, Policy.ReadWrite.ConditionalAccess
Application	Policy.ReadWrite.SecurityDefaults, Policy.ReadWrite.ConditionalAccess

	Policies
	Disable Security Defaults
Azure AD Admin Roles	
Conditional Access Administrator	Success
Directory Writers	Success
Exchange Administrator	Failure
Global Administrator	Success

- We found that no application permission can disable Security Defaults
- Only roles granted appropriate delegated permissions **(excluding Directory.AccessAsUser.All)** can perform this action

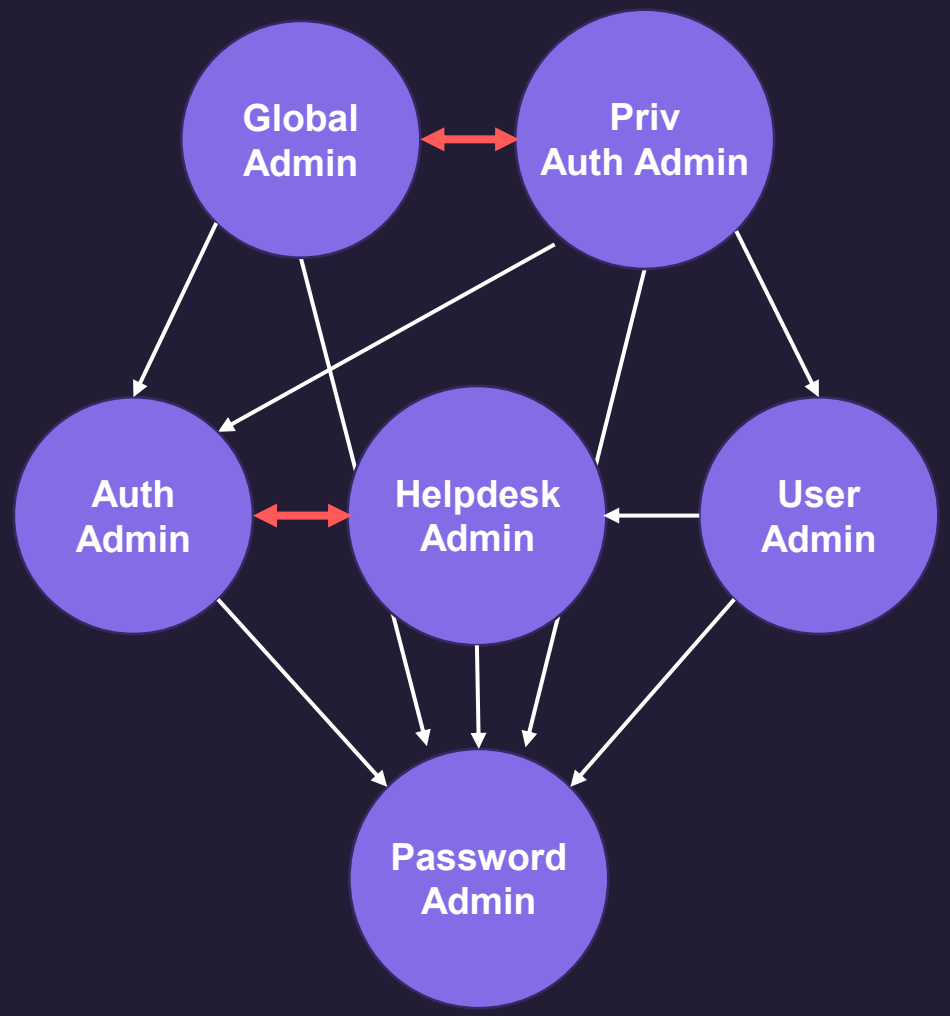
In some cases, the outcome of operation depends on the **target object**

Attack path prevention system

- Entra ID has a built-in system to protect against the emergence of attack paths, particularly around password reset privileges

This is a privileged role. Users with this role can change passwords, invalidate refresh tokens, create and manage support requests with Microsoft for Azure and Microsoft 365 services, and monitor service health. Invalidating a refresh token forces the user to sign in again. Whether a Helpdesk Administrator can reset a user's password and invalidate refresh tokens depends on the role the user is assigned. For a list of the roles that a Helpdesk Administrator can reset passwords for and invalidate refresh tokens, see Who can reset passwords.

Password reset privileges in Entra ID

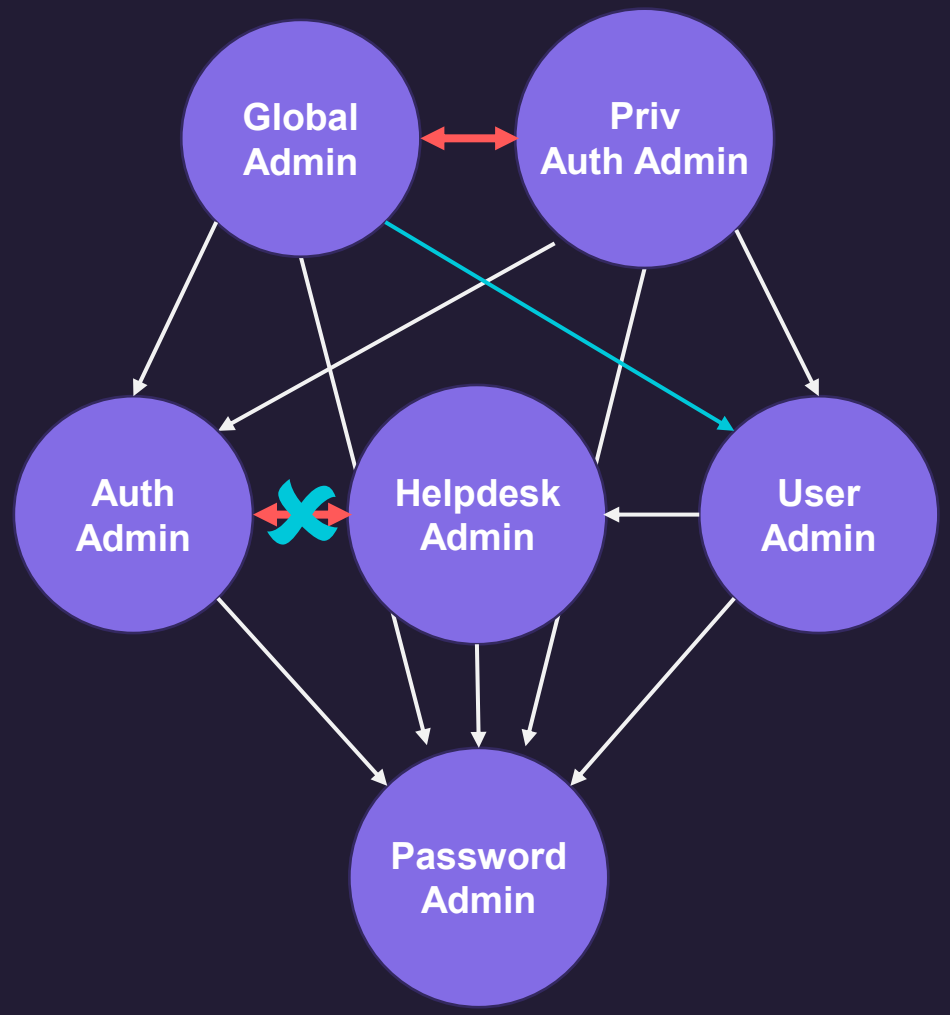


- **Built-In Privilege Escalation Prevention System**
  - Whether Administrators can reset a user's password **depends on the role the user is assigned**
- Andy Robbins enumerate all the different possibilities for password reset privileges in Entra ID **in 2021**

	Can a User with Role in Column A reset a password for a user with a Role in Row 2?													
	(No Role)	Global Administrator	Privileged Authentication Administrator	Helpdesk Administrator	Authentication Administrator	User Administrator	Password Administrator	Directory Readers	Guest Inviter	Message Center Reader	Privileged Role Administrator	Reports Reader	Groups Administrator	(Any Other Role)
Global Administrator	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Privileged Authentication Administrator	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Helpdesk Administrator	Yes	No	No	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	No	No
Authentication Administrator	Yes	No	No	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	No	No
User Administrator	Yes	No	No	Yes	No	Yes	No	Yes	Yes	Yes	No	Yes	No	No
Password Administrator	Yes	No	No	No	No	No	Yes	Yes	Yes	No	No	No	No	No

Azure Privilege Escalation via Service Principal Abuse

Who can reset passwords (Official Version)



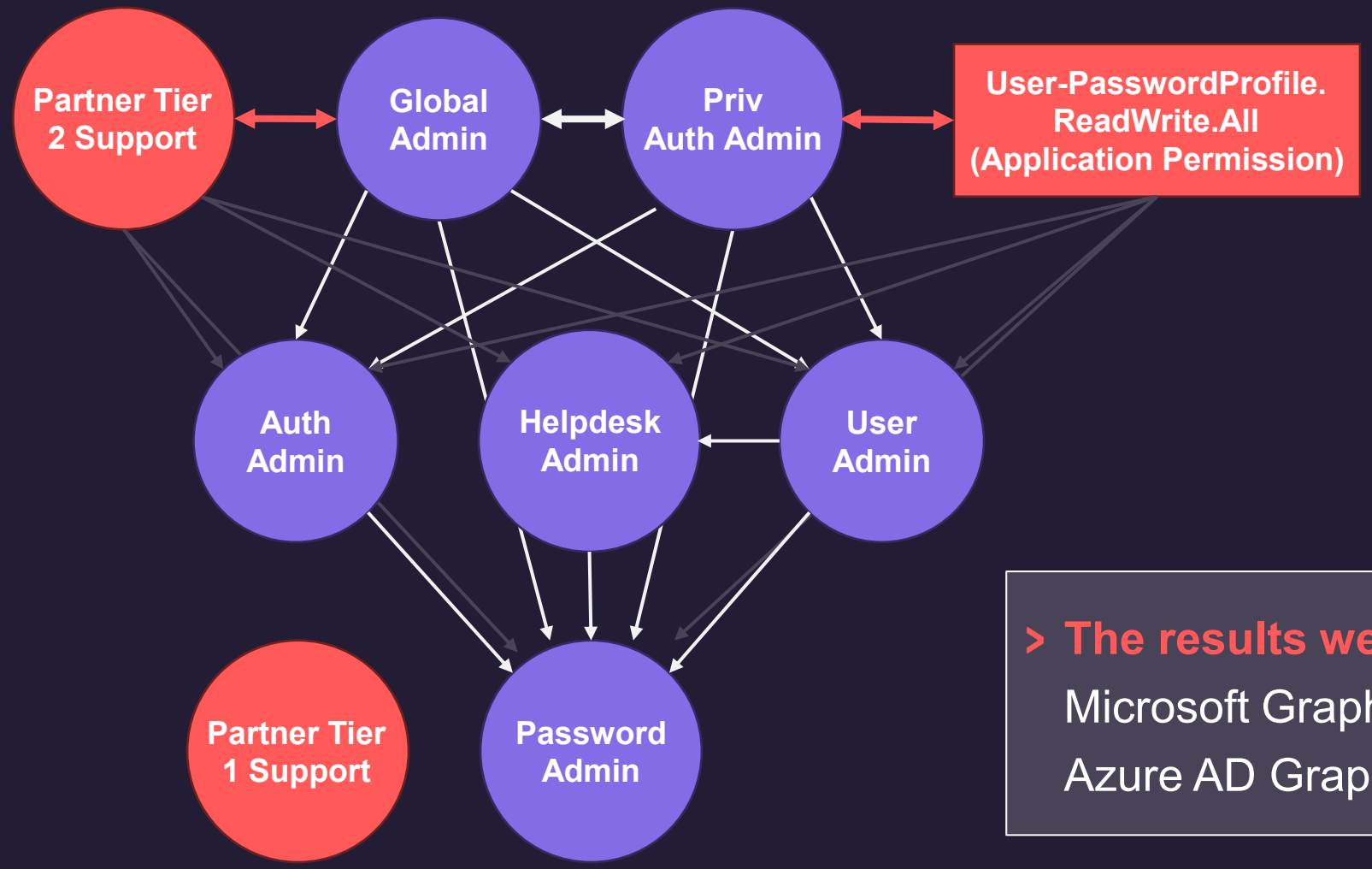
> Microsoft updated the documentation, with the latest revision dated October 2024

Role that password can be reset	Password Admin	Helpdesk Admin	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 roles and permissions in Microsoft Entra ID (preview)

CYCRAFT

Who can reset passwords (Our Version)



> **The results were consistent** across Microsoft Graph API (v1 and beta) and Azure AD Graph (1.61 and 1.61-internal)

Misalignment between the documentation

- Microsoft Graph permissions reference is correct
- The Microsoft Graph API documentation is out of dated

User-PasswordProfile.ReadWrite.All

Expand table

Category	Application	Delegated
Identifier	cc117bb9-00cf-4eb8-b580-ea2a878fe8f7	56760768-b641-451f-8906-e1b8ab31bca7
DisplayText	Read and write all password profiles and reset user passwords	Read and write password profiles and reset user passwords
Description	Allows the app to read and write password profiles and reset passwords for all users, without a signed-in user.	Allows the app to read and write password profiles and reset passwords for all users, on behalf of the signed-in user.
AdminConsentRequired	Yes	Yes

authenticationMethod: resetPassword

Article • 09/10/2024 • 13 contributors

Feedback

Permission type	Least privileged permissions	Higher privileged permissions
Delegated (work or school account)	UserAuthenticationMethod.ReadWrite.All	Not available.
Delegated (personal Microsoft account)	Not supported.	Not supported.
Application	Not supported.	Not supported.

authenticationMethod: resetPassword

Pyark (Scheduled for release before September)

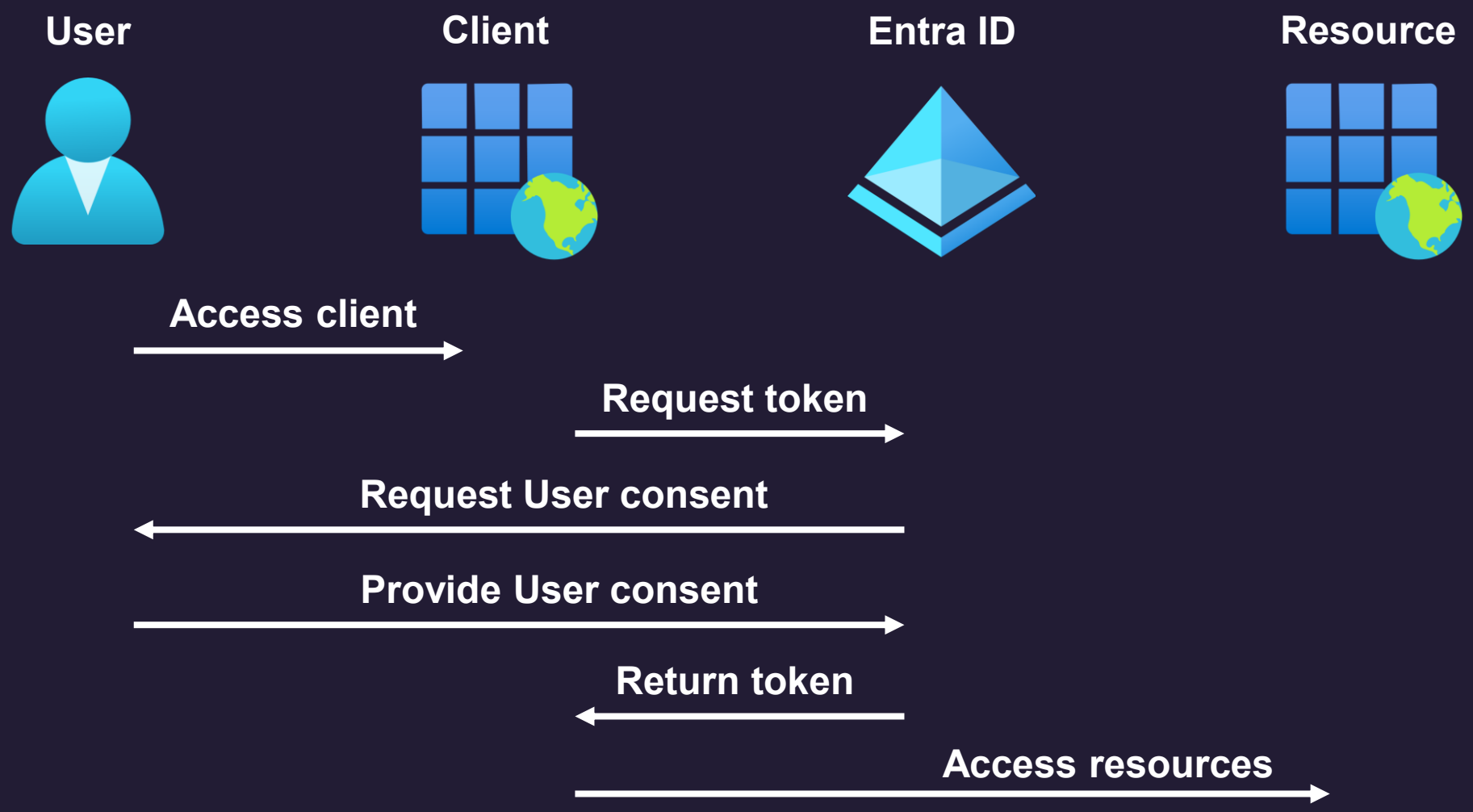
- **Test Identity**
  - Service Principal
  - User
    - Disable Security Default
    - Assign known abusable delegated permissions to the Microsoft Graph Command Line Tools
- **Support API Type**
  - Microsoft Graph API
  - Azure AD Graph API
    - Scheduled for retirement in July 2025
  - Ibiza IAM API (undocumented API)
    - Extension API for Web Portal
    - **Nodoc** project provides thorough documentation of the API

Minor permission misalignments were observed between different APIs

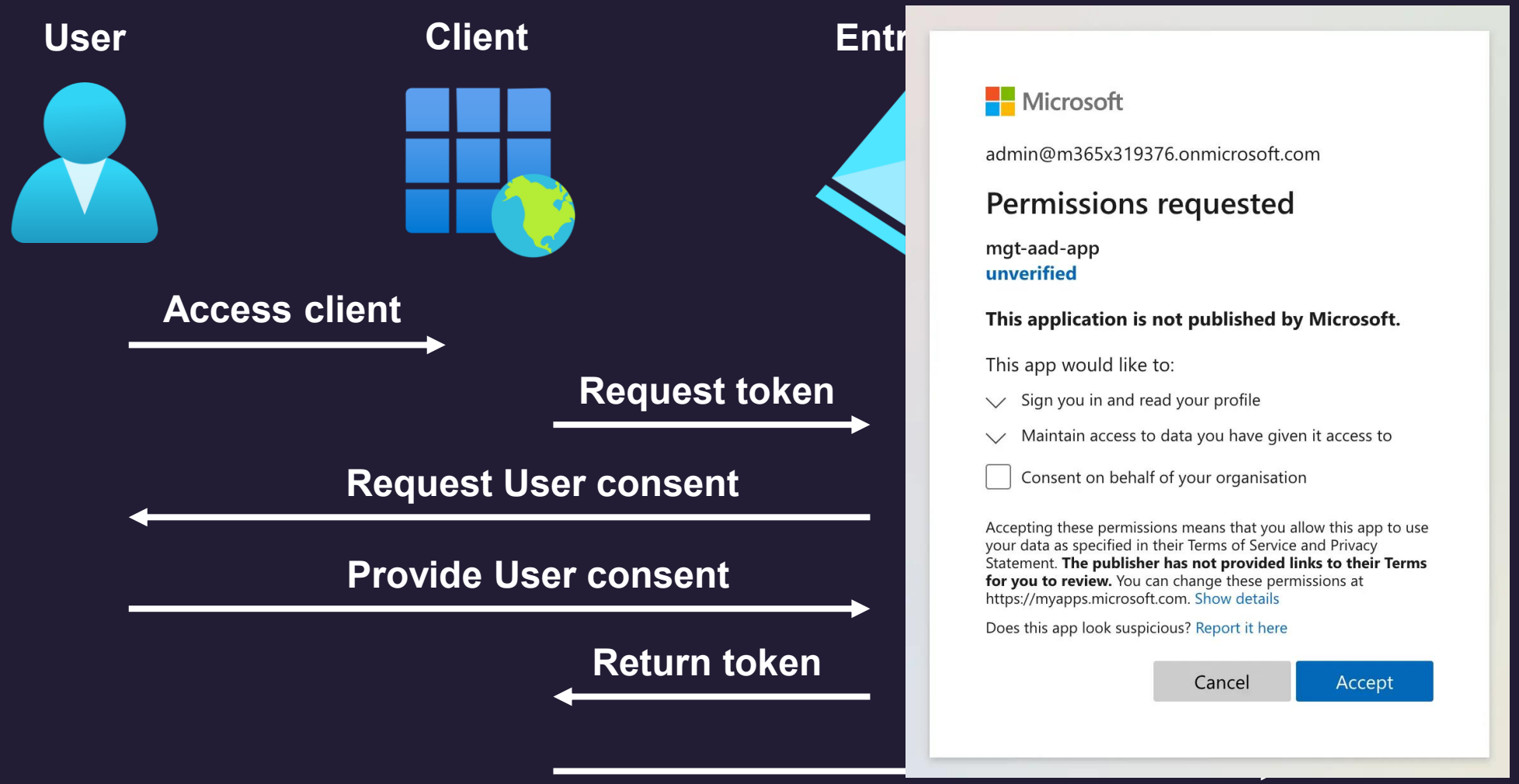
No newly abusable Entra Roles were identified

Why can operations be performed through the Azure portal without consent

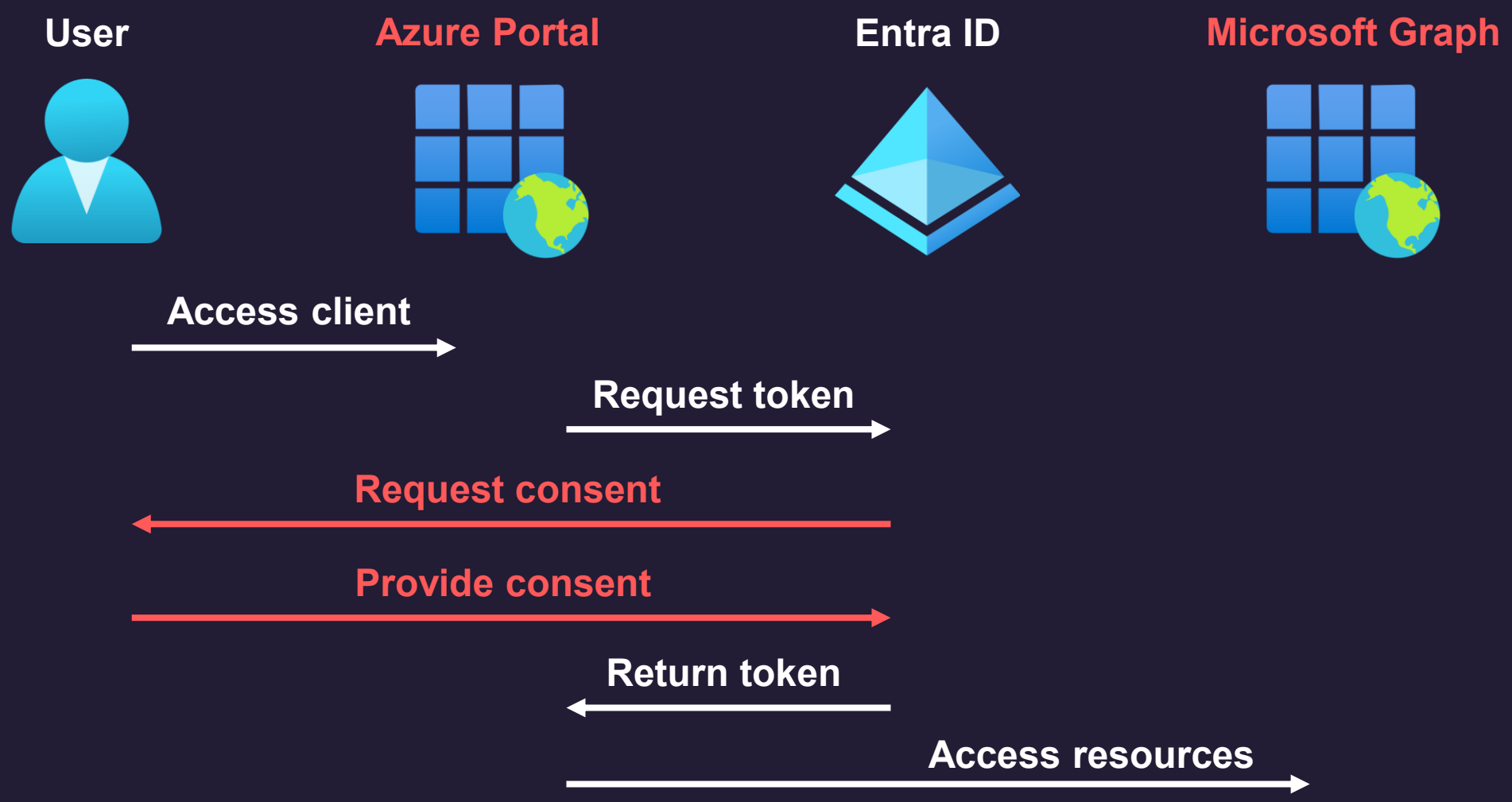
Authorization model in Entra ID



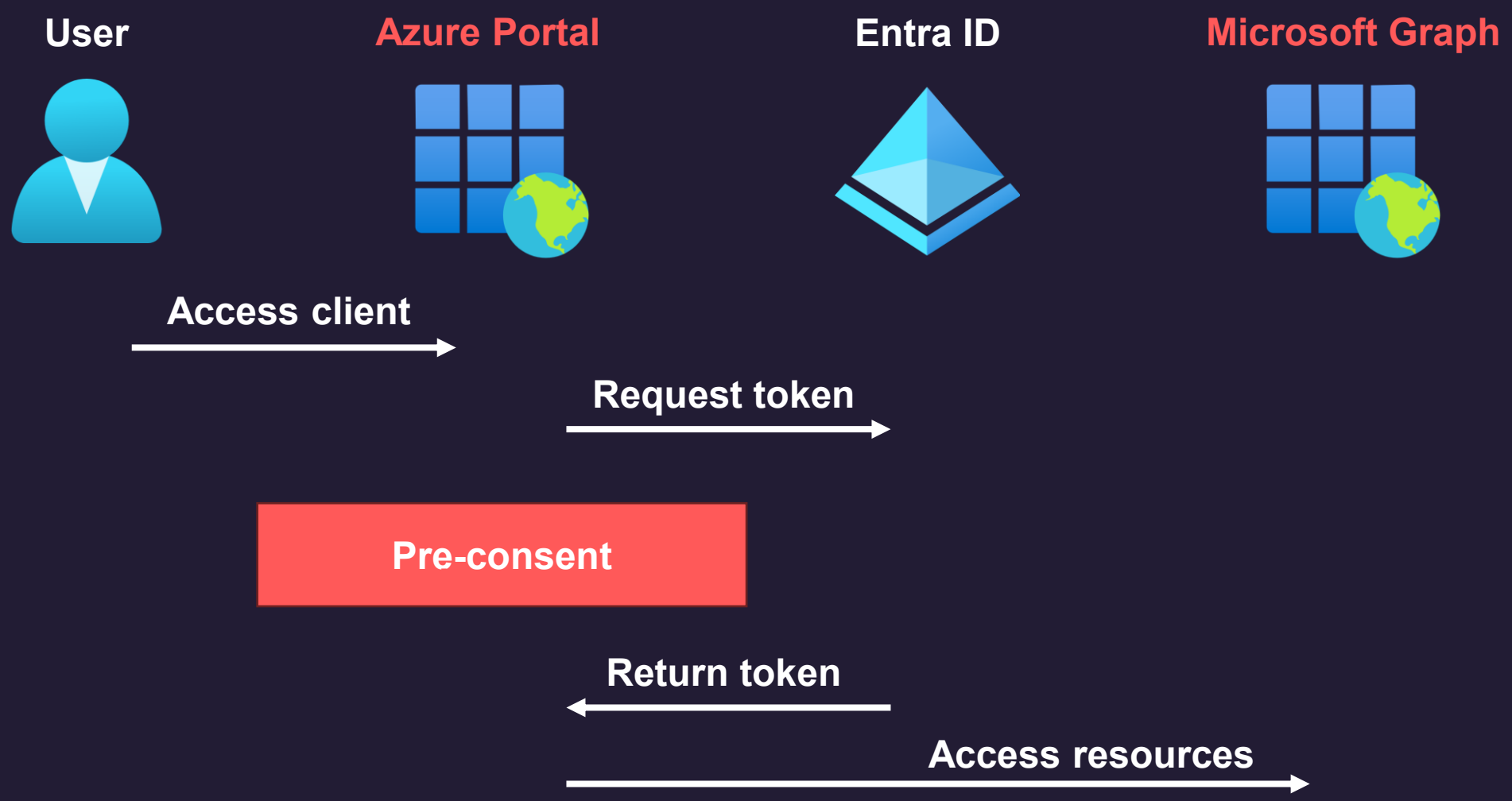
Authorization model in Entra ID



Pre-consented Delegated Permissions



Pre-consented Delegated Permissions



Does the Azure Portal pre-consent to all Microsoft Graph permissions?

Token Redemption flow behind Azure Portal

Name | Headers | Payload | Preview | Response | Initiator | Timing

getAuthTo...
token
token
token
token

Form Data | View source | View URL-encoded

client_id: c44b4083-3bb0-49c1-b47d-974e53cbdf3c
scope: openid profile offline_access
code: [redacted]

Headers | Payload | Preview | Response | Initiator | Timing | Cookies

```
1 {
  "token_type": "Bearer",
  "scope": "https://management.core.windows.net//user_impersonation
  "expires_in": 5117,
  "ext_expires_in": 5117,
  "access_token": "[redacted]
  "refresh_token":
  "id_token": "eyJ0[redacted]
```

Name | Head | Azure Portal | esponse | Initiator | Timin

getAuthTo...
token
token
token
token

Form Dat | View URL-encoded

client_id: c44b4083-3bb0-49c1-b47d-974e53cbdf3
scope: c44b4083-3bb0-49c1-b47d-974e53cbdf3
grant_type: refresh_token
client_info: 1

Headers | Payload | Preview | Response | Initiator | Timing | Cookies

```
1 {
  "token_type": "Bearer",
  "scope": "c44b4083-3bb0-49c1-b47d-974e53cbdf3c/Organization.Read.All
  "expires_in": 4298,
  "ext_expires_in": 4298,
  "access_token":
  "refresh_token":
  "id_token": "ey[redacted]
```

Name | Headers | Payload | Preview | Response | Initiator | Timing

getAuthTo...
token
token
token
token

Form Data | View source | View URL-encoded

client_id: c40dfea8-483f-469b-aafe-642149115b3a
scope: https://graph.microsoft.com//.default openi
grant_type: refresh_token

Headers | Payload | Preview | Response | Initiator | Timing | Cookies

```
1 {
  "token_type": "Bearer",
  "scope": "email openid profile https://graph.microsoft.com//AdministrativeUnit.
  "expires_in": 5275,
  "ext_expires_in": 5275,
  "access_token": [redacted]
  "refresh_token":
```

CYCRAFT

Token Redemption flow behind Azure Portal


```
{
  "aud": "https://management.core.windows.net/",
  "iss": "https://sts.windows.net/5eecfe8b-9e19-477f-9a39-1652d4fcfe7
9/",
  "exp": 1723465285,
  "acr": "1",
  "aio": "AVQAq/8XAAAA/hx9aHLBbNcb+VxfvQrr8l8KJAajA1WQUT7/y09GmcVKc
cQuRyRdiG716fR/uGl+qdFRgSWJLHMtBWn7ceeeMD2rACrfbUGNLmVSQXk=",
  "amr": [
    "pwd",
    "mfa"
  ],
  "appid": "c44b4083-3bb0-49c1-b47d-974e53cbdf3c",
  "appidacr": "0",
  "groups": [
    "5bad188f-359f-47cc-b933-393f5f2bd607"
  ],
  "idtyp": "user",
  "ipaddr": "1.162.141.66",
  "name": "lumian",
  "oid": "9106e5b3-1818-41b9-a6ba-aece23fd14ca",
  "puid": "10032003AE412380",
  "rh": "0.ASsAi_7sXhmef0eaORZS1Pz-eUZIf3kAutdPukPawfj2MBPCAEg.",
  "scp": "user_impersonation",
  "sub": "9InT1Vi-989L_2tKkga1UfKIO6hWPJ-kbNfrosUSOog",
  "tid": "5eecfe8b-9e19-477f-9a39-1652d4fcfe79",
  "unique_name": [redacted],
  "upn": [redacted],
  "uti": "WcDLUXojHEunDXHHOGdbAA",
```


Azure Resource Manager

Change Resource of the Request

Change Client id of the Request

Microsoft Graph


```
{
  "aud": "https://graph.microsoft.com/",
  "iss": "https://sts.windows.net/5eecfe8b-9e19-477f-9a39-1652d4fcfe7
9/",
  "iat": 1723459868,
  "nbf": 1723459868,
  "exp": 1723465444,
  "acr": 0,
  "acrs": "1",
  "aio": "AVQAq/8XAAAAiwcEdAUb+fcFfeNjOecwod4/vtuKVehoTyQm8vVUGoNvpivTg
0KI/jqQ30LXexd0gJoKrkBvq9aj/D9vFRlnZ1FlpO7Vmdd5qC8EksoTpg0=",
  "amr": [
    "pwd",
    "mfa"
  ],
  "app_displayname": "Microsoft_AAD_Devices",
  "appid": "c40dfea8-483f-469b-aafe-642149115b3a",
  "appidacr": "0",
  "idtyp": "user",
  "ipaddr": "1.162.141.66",
  "name": "lumian",
  "oid": "9106e5b3-1818-41b9-a6ba-aece23fd14ca",
  "rh": "3",
  "puid": "10032003AE412380",
  "rh": "0.ASsAi_7sXhmef0eaORZS1Pz-QMAAAAAAAAAwAAAAAAAAADCAEg.",
  "scp": "AdministrativeUnit.Read.All AdministrativeUnit.ReadWrite.All
AuditLog.Read.All BitlockerKey.Read.All Device.Read.All DeviceLocalCre
dential.Read.All DeviceManagementManagedDevices.Read.All Directory.Wri
te.Restricted email GroupMember.Read.All openid Policy.Read.All Polic
y.ReadWrite.Authorization Policy.ReadWrite.DeviceConfiguration profile
User.ReadBasic.All",
  "signin_state": [
```


CYCRAFT

Special Redemption Flow (BrkRefresh)


```python
header   = {'Content-Type': 'application/x-www-form-urlencoded',
            'origin':'https://portal.azure.com'}                                  #REQUIRED

body   = {
    "client_id": client_id,
    "scope": scope,
    "grant_type": "refresh_token",
    "refresh_token": refresh_token,
    "brk_client_id": "c44b4083-3bb0-49c1-b47d-974e53cbdf3c",                      #REQUIRED
    "redirect_uri": "brk-c44b4083-3bb0-49c1-b47d-974e53cbdf3c://portal.azure.com"    #REQUIRED
    }

response = requests.post(f'https://login.microsoftonline.com/{tenant_id}/oauth2/v2.0/token',
                         headers = header,
                         data=body)
```

Family of Client ID (FOCI)

> "FUTURE SERVER WORK WILL ALLOW CLIENT IDS TO BE GROUPED ON THE SERVER SIDE IN A WAY WHERE A RT FOR ONE CLIENT ID CAN BE REDEEMED FOR A AT AND RT FOR A DIFFERENT CLIENT ID AS LONG AS THEY'RE IN THE SAME GROUP. THIS WILL MOVE US CLOSER TO BEING ABLE TO PROVIDE SSO-LIKE FUNCTIONALITY BETWEEN APPS WITHOUT REQUIRING THE BROKER (OR WORKPLACE JOIN)."

We then found references in the source code calling refresh tokens issued to FOCI clients "family refresh tokens" (or FRTs). Based on developer remarks, it appears there is only one family ID currently in use at Microsoft.

In MSRC submission VULN-057712, Microsoft confirmed that FOCI and family refresh toke

Not Exactly

software feature. Microsoft engineering provided a thoughtful (and quite lengthy) response describing the origins of FOCI and its threat model, which confirmed the findings from this research. According to Microsoft, FOCI was designed to support pseudo single sign-on (SSO) functionality for Microsoft mobile applications. FOCI mirrors the behavior of mobile operating systems that store authentication artifacts (such as refresh tokens) in a shared token cache with other applications from the same software publisher.

GraphPreConsentExplorer

Load YML File

Applications: 9 / Unique MS Graph permissions: 51

Microsoft_AAD

Enabled only | With permission only | Brk Refresh Flow | Filter by FOCI

App Name	Client ID	Enabled	Graph API Permissions	Auth Flow	FOCI
Microsoft_AAD_UsersAndTenants	f9885e6e-6f74-46b3-b595-350157a27541	True	AdministrativeUnit.ReadWrite.All, AuditLog.Read.All, Directory.AccessAsUser.All, Directory.Write.Restricted, email, EventListener.ReadWrite.All, IdentityRiskyUser.Read.All, openid, Organization.Read.All, Policy.ReadWrite.Authorization, profile, PublicKeyInfrastructure.ReadWrite.All, User.EnableDisableAccount.All, User.ReadWrite.All	BrkRefresh	False
Microsoft_AAD_Devices	c40dfea8-483f-469b-aafe-642149115b3a	True	AdministrativeUnit.Read.All, AdministrativeUnit.ReadWrite.All, AuditLog.Read.All, BitlockerKey.Read.All, Device.Read.All, DeviceLocalCredential.Read.All, DeviceManagementManagedDevices.Read.All, Directory.Write.Restricted, email, GroupMember.Read.All, openid, Policy.Read.All, Policy.ReadWrite.Authorization, Policy.ReadWrite.DeviceConfiguration, profile, User.ReadBasic.All	BrkRefresh	False

Takeaways

- The Microsoft documentation may lack clarity or reflect outdated information
  - Analysis beyond the official documentation is required
- Service-based abuse testing may result incomplete coverage
- **Brkfresh** allows token redemption across client apps in the group
- Additional MS Graph API permissions to be aware of:
  - Delegated Permission: **Directory.AccessAsUser.All**
  - Application Permission: **User-PasswordProfile.ReadWrite.All**

ACKNOWLEDGEMENT

- **@_wald0** – BARK
- **@cnotin** – entra-id-federation-abuse-research-required-roles
- **@x_delfino** – nodoc
- **@ZH54321** – GraphPreConsentExplorer

Q & A

You can find me at **@iflywithoutwind**