

## BYO IDP in Entra ID

Persisting and bypassing MFA with OIDC based protocols

## About me



- Dirk-jan Mollema
- Lives in The Netherlands
- Hacker / Researcher / Founder / Trainer @ Outsider Security
- Talks at Black Hat / Def Con / BlueHat / Troopers / x33fcon
- Author of several Active Directory and Entra ID tools
  - mitm6
  - Ldapdomaindump
  - adidnsdump
  - BloodHound.py
  - ntlmrelayx / krbrelayx
  - ROADtools

Socials Blog/talks: Twitter/X: BlueSky:

dirkjanm.io @\_dirkjan @dirkjanm.io

## Talk agenda

- What is OIDC and why should we care
- OIDC and federated credentials
- Entra External Authentication Methods
- Conditional Access custom controls
- Detection opportunities and challenges

### Entra ID

- Microsoft's Identity-as-a-service platform
- Formerly called Azure AD
- Practically handles all cloud-based sign-ins in modern enterprises

## Why OpenID connect

- Most of the time we want to have access tokens / bearer tokens
- They give us access to data
- Focus often on Microsoft 365 native apps
- Little interest in how the actual validation of these tokens works

## Why OpenID connect

- Whereas access tokens provide **authorization** for resources, ID tokens provide **authentication** to clients
- Web and native apps can use ID tokens to authenticate a user
- Used extensively for SSO with Entra ID

## OIDC technical bits

- Based on OAuth2
- The Authorization Server (often Entra ID) will issue an ID token during the familiar OAuth flows
- ID token requested by openid scope
- Main difference with access token:
  - Access token is intended for the Resource Provider (upstream API such as Microsoft Graph) to authorize the user
  - ID token is intended for the client to authenticate the user

# Federated credentials

## Application credentials

💡 testapp5   Certificates & secrets 👒 …						
	🔗 Got feedback?					
<ul> <li>Overview</li> <li>Ouickstart</li> </ul>	Credentials enable confidential ap	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.				
Integration assistant	scheme). For a higher level of assu					
<ul> <li>Diagnose and solve problems</li> <li>Manage</li> </ul>	() Application registration certif	() Application registration certificates, secrets and federated credentials can be found in the tabs below.				
<ul> <li>Branding &amp; properties</li> <li>Authentication</li> <li>Certificates &amp; secrets</li> </ul>	Certificates (0) <b>Client secrets (0)</b> 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.			n be referred to as application password.		
<ul> <li>Token configuration</li> <li>API permissions</li> </ul>	H New client secret Description	Expires	Value ①	Secret ID		
<ul> <li>Expose an API</li> <li>App roles</li> <li>Owners</li> </ul>	No client secrets have been crea	ted for this application.				
Roles and administrators						
<ul> <li>Manifest</li> <li>Support + Troubleshooting</li> <li>New support request</li> </ul>						

## Application credentials

- Client secrets or passwords
  - Use the client secret itself to auth
- Certificates + private key
  - Use signed assertion to authenticate
- Federated Credentials
  - Allow another IDP to authenticate?

### Federated credentials

- Federation = establishing a trust relationship between an application (Service Provider or Resource Provider) and an Identity Provider so that the IdP can authenticate users to your app.
- Federation = making something else responsible for your authentication and trusting they do their job properly.
- Best known example: AD FS
- Essentially trusting another IdP (not Entra ID) to issue tokens so we can get tokens from the IdP we want (Entra ID)

## Configuring federated credentials

#### Add a credential

Allow other identities to impersonate this application by establishing a trust with an external OpenID Connect (OIDC) identity provider. This federation allows you to get tokens to access Microsoft Entra ID protected resources that this application has access to like Azure and Microsoft Graph. Learn more

 $\sim$ 

Federated credential scenario \*

GitHub Actions deploying Azure resources

Connect your GitHub account

Please enter the details of your GitHub Actions workflow that you want to connect with Microsoft Entra ID. These values will be used by Microsoft Entra ID to validate the connection and should match your GitHub OIDC configuration. Issuer has a limit of 600 characters. Subject Identifier is a calculated field with a 600 character limit.

Issuer (i)	https://token.actions.githubusercontent.com	
	Edit (optional)	
Organization *	GitHub organization name	
Repository *	GitHub repository name	
Entity type *	Branch	$\vee$
GitHub branch name *	Value	
Subject identifier (i)	repo:{Organization}/{Repository}:ref:refs/heads/{Branch}	
	This value is generated based on the GitHub account details provided. Edit (optional)	

## Custom providers

#### Add a credential

Allow other identities to impersonate this application by establishing a trust with an external OpenID Connect (OIDC) identity provider. This federation allows you to get tokens to access Microsoft Entra ID protected resources that this application has access to like Azure and Microsoft Graph. Learn more

 $\sim$ 

Other issuer		

#### Connect your account

Enter the details of the account that you want to connect with Microsoft Entra ID. These values will be used by Microsoft Entra ID to validate the connection.

lssuer *	Issuer URL (Limit of 600 characters)	
Туре 🛈	• Explicit subject identifier O Claims matching expression (Preview)	
Value * 🛈	Subject (Limit of 600 characters)	
	S The subject identifier field is required	

### Federated credentials auth

#### • Uses OAuth2 client credentials flow with signed assertion

#### Third case: Access token request with a federated credential



### Federated credentials protocol





#### roadoidc

- Minimalistic OIDC implementation
- Can be hosted on Azure App services or Azure Blob storage
  - Azure Blob storage hosts static files, suitable for federated credentials but not for advanced scenarios we cover later
- Tokens can be requested with roadtx



## Authenticating with federated creds

Certificates (0) Client secrets (0) Fe	ederated credentials (1)				
Allow other identities to impersonate this application by establishing a trust with an external OpenID Connect (OIDC) identity provider. This federation allows you to get tokens to access Microsoft Entra ID protected resources that this application has access to like Azure and Microsoft graph. Learn more					
+ Add credential					
Name	Description	Subject identifier or claims mate	ching expression		
roadoidc	Totally legit identity provider	testapp5	Ē		

(ROADtools) → roadoidc git:(master) × roadtx federatedappauth -c 2dfa5a93-3400-4b81-b6d4-2123c6c1ae6e --cert-pem roadoidc.pem --key-pem roadoidc.key --subject testapp5 -t iminyour.cloud --issuer https://roadoidcblob.blob.core.windows.net/containername -s https://graph.microsoft.com/.default Requesting token with scope https://graph.microsoft.com/.default Tokens were written to .roadtools\_auth

## What's the point

- Mainly OPSEC
  - Adding client secrets and certificates to apps is a well-known technique and included in many detection playbooks.
  - Federated credentials are less well known by defenders and may not be spotted.

#### Adding client secret

TimeGenerated [UTC] $\uparrow \downarrow$		ResourceId	OperationName	OperationVersion	Category
>	6/11/2025, 8:03:07.864 AM	/tenants/6287f28f-4f7f-4322-9	Update application	1.0	ApplicationManagement
>	6/11/2025, 8:03:07.863 AM	/tenants/6287f28f-4f7f-4322-9	Update application – Certificates and secrets management	1.0	ApplicationManagement
>	6/11/2025, 8:03:07.799 AM	/tenants/6287f28f-4f7f-4322-9	Update service principal	1.0	ApplicationManagement
Adding federated credentials					
>	6/11/2025, 7:57:51.389 AM	/tenants/6287f28f-4f7f-4322-9	Update application	1.0	ApplicationManagement
>	6/11/2025, 7:57:51.328 AM	/tenants/6287f28f-4f7f-4322-9	Update service principal	1.0	ApplicationManagement

### Federated credentials in Azure

- Federated credentials exist on User Managed Identities
- Normally, managed identities can only be accessed via resources they are linked to
  - Via Metadata endpoint on Virtual Machines, Logic Apps, etc
  - Gives out only access tokens, no long-term persistence without resource access
- Federated credentials allow for persistent access
  - Permanent credentials that can be used at any time
  - Can be used outside of Azure

### Federated credentials on MI

#### Home > automation-mis

Anaged Identity	is   Fe	derated credentials $\Rightarrow$				
🔎 Search	× «	Federated credentials				
🚷 Overview		Configure an identity from an external OpenID Connect Provider to get token	s as this managed identity to access Microsoft Entra ID protected services.Learn more about ho	w to create an identity from an external OpenID		
Activity log						
Access control (IAM)		+ Add Credential				
🇳 Tags		2 of 20 configured				
<b>†</b> Azure role assignments						
Associated resources (preview	v)	Name ↑	Issuer	Subject Identifier		
🕂 Resource visualizer		roadoidc	https://roadoidcapp.azurewebsites.net	testapp		
$\checkmark$ Settings		bla	https://roadoidcblob.blob.core.windows.net/containername	bla		
$   \   \rho \ $ Federated credentials	☆					
Properties						
🔒 Locks						
✓ Monitoring						

Advisor recommendations

## **OIDC in External Auth Methods**

## **External Authentication Methods**

- New-ish feature (May 2024) that makes it possible to use external MFA providers.
- Alternative for Entra ID native MFA.
- Uses OpenID Connect to trust authentication claims from IdPs



Reference: https://techcommunity.microsoft.com/blog/microsoft-entra-blog/public-preview-external-authentication-methods-in-microsoft-entra-id/4078808

## Test setup

• Shout-out to DUO for free tier!

ı. cısco	DU÷				Q Search
≡	Collapse		← Applications Microsoft Entra II	D: External Authentication Meth	ods
	Home		See the Duo Entra ID External	Authentication Method Documentation 🗗 to integrate Du	o with Microsoft Entra ID.
2=	Users	>	<b>Basic Configuration</b>		
	Devices	>			
1	Policies	>	Application name *	Microsoft Entra ID: External Authentication Methods	
	Applications	>			
	Reports	>	Application Type	Microsoft Entra ID: External Authentication Methods	
	Billing	>	User access	Disable for all users	
ø	Settings			Enable for all users	
			Details	aqainst Microsoft Entra ID Tenant ID: 6287f28f-4f7f-432	2-9651-a8697d8fe1bc
			Use the information below to c	create an external authentication method in your Microso	ft Entra ID tenant.
				,	
			Name	Cisco Duo	Сору
			Client ID	YXBpLWEyYTM0NDFkLmR1b3NIY3VyaX	Сору
			Discovery Endpoint	https://eu-west.azureauth.duosecurity.c	Сору
			App ID	0d187fb5-d1d7-45e9-b567-b270338ad	Сору

## Test setup – configure EAM

Home > iminyourcloud   Security > Security   Authentication methods > Authentication methods   Policies >					
<b>Cisco Duo 2</b> External Authentication Methods (Preview)					
🔟 Delete					
Enable and target Configure					
Method Properties					
Your provider will give you the name, clie	ent ID, discovery endpoint, and app ID for the external authentication method.				
Name *	Cisco Duo 2				
	The provider name cannot be changed.				
Client ID *	YXBpLWEyYTM0NDFkLmR1b3NIY3VyaXR5LmNvbTpESTNVWEc5M0xCQzRXSk8				
Discovery Endpoint *	https://eu-west.azureauth-duosecurity.com/.well-known/openid-configuration				
App ID *	0d187fb5-d1d7-45e9-b567-b270338ad8a5				
Request admin consent	Admin consent granted				



## EAM request

#### Request

🗞 🚍 \n ≡ Pretty Raw Hex 1 POST /authorization HTTP/1.1 2 Host: eu-west.azureauth.duosecurity.com 3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86 64; rv:136.0) Gecko/20100101 Firefox/136.0 4 Accept: text/html,application/xhtml+xml,application/xml;g=0.9,\*/\*;g=0.8 5 Accept-Language: en-US, en; q=0.5 6 Accept-Encoding: gzip, deflate, br 7 Referer: https://login.microsoftonline.com/ 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 2424 Origin: https://login.microsoftonline.com 11 Upgrade-Insecure-Requests: 1 12 Sec-Fetch-Dest: document 13 Sec-Fetch-Mode: navigate 4 Sec-Fetch-Site: cross-site 15 Priority: u=0, i 16 Te: trailers 17 Connection: keep-alive 18 19 scope=openid&response mode=form post&response type=id token&client id= YXBpLWEyYTM0NDFkLmR1b3N1Y3VyaXR5LmNvbTpESTNVWEc5M0xCQzRXSk800Ex0UQ%3D%3D&redirect uri= https%3A%2F%2Flogin.microsoftonline.com%2Fcommon%2Ffederation%2Fexternalauthprovider&claims= %7B%22id token%22%3A%7B%22amr%22%3A%7B%22essential%22%3Atrue%2C%22values%22%3A%5B%22face%22%2C%22fido%22%2C %22fpt%22%2C%22hwk%22%2C%22iris%22%2C%22otp%22%2C%22tel%22%2C%22pop%22%2C%22retina%22%2C%22sc%22%2C%22sms%2 2%2C%22swk%22%2C%22vbm%22%2C%22bio%22%5D%7D%2C%22acr%22%3A%7B%22essential%22%3Atrue%2C%22values%22%3A%5B%22 possessionorinherence%22%5D%7D%7D%7D&nonce= AWABEGEAAAADAOz BODO 7pmdPuilwSOnbEHB21RGEuibpS2erMt7BvN-uBXaemX7 7taall TL pBec Hr7SG5YOA6oobyoOmO7i2sOKRrk aAA&id token hint= eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiIsImtpZCI6IkNOdjBPSTNSd3FsSEZFVm5hb01Bc2hDSDJYRSJ9.eyJhdWQi0iIwZDE4N2ZiNS 1kMWQ3LTQ1ZTktYjU2Ny1iMjcwMzM4YWQ4YTUiLCJpc3MiOiJodHRwczovL2xvZ2luLm1pY3Jvc29mdG9ubGluZS5jb20vNjI4N2YyOGYtN GY3Zi00MzIyLTk2NTEtYTq2OTdk0GZLMWJjL3YyLjAiLCJpYXQi0jE3NDk3MjcyMTIsIm5iZiI6MTc0OTcyNzIxMiwiZXhwIjoxNzQ5NzI4 MTEyLCJuYW1lIjoiZHVvMyIsIm9pZCI6IjE4Y2JkZjFiLWU2OTQtNGU1MC1hZTU2LTkxNTYyMzU1NzM5ZiIsInByZWZlcnJlZF91c2VybmF tZSI6ImR1bzNAaW1pbnlvdXIuY2xvdWQiLCJzdWIiOiJvZ0pHcUZoUENGZjMvb3gxUnZWTmVfUk9pV1dfazcyNTFZSlBlYVRvak80IiwidG lkIjoiNjI4N2YyOGYtNGY3Zi00MzIyLTk2NTEtYTg2OTdkOGZlMWJjIiwidXBuIjoiZHVvM0BpbWlueW91ci5jbG91ZCIsInV0aSI6IlRxY nVXRXZUMVVLZmJjNTdkQzhHQUEiLCJ2ZXIiOiIyLjAifQ.A25QsnXRxHtbcdXo HMx34eKL6eQobfM8n8kf4K W3fu7B6ysJ8G4bcvhX9Eg IlBX5cNgaZ7BPwm5oV7dTdmKIPvHsNfZ1DxVt373 SG7 ccQTFg824z9mm cndRPA7vtM9aruO-rJlvw2iKxrCZTzXxNQiaHtmlFiePhSvO

RDUcis0pq3L-Gt5qxvNMcjqliW-8KH -MVGlVkXc VaclosvOpUuOv3xEB3R7UAI65ldR-YcT2T9FlEqTh73rKSftsTbrhMRSQP8I4B1CBi Q9VmW2i3ty05VRWuLvy2EwjLUkhDCMmT2SDRoQ9rAY5IYkqV0KbDpoljYw6m4hBsEug&client-request-id= /28085a8-3008-4235-a494-081390818e55&State=

## ID token signed by Entra ID

HEADER: ALGORITHM & TOKEN TYPE	
{ "typ": "JWT", "alg": "RS256", "kid": "CNv00I3RwqlHFEVnaoMAshCH2XE" }	
PAYLOAD: DATA	
<pre>{     "aud": "0d187fb5-d1d7-45e9-b567-b270338ad8a5",     "iss": "https:// login.microsoftonline.com/6287f28f-4f7f-4322-9651- a8697d8fe1bc/v2.0",     "iat": 1749727212,     "nbf": 1749727212,     "nbf": 1749727212,     "exp": 1749728112,     "name": "duo3",     "oid": "18cbdf1b-e694-4e50-ae56-91562355739f",     "preferred_username": "duo3@iminyour.cloud",     "    "    "    "    "    "    "</pre>	
"sub": "ogJGqFhPCFf30ox1RvVNe_R0iWW_k7251YJPeaToj04", "tid": "6287f28f-4f7f-4322-9651-a8697d8fe1bc", "upn": "duo3@iminyour.cloud", "uti": "TqbuWEvT1UKfbc57dC8GAA",	

"ver": "2.0

## EAM return result

#### Request

Pretty Raw Hex

🗞 🚍 \n ≡

1 POST /common/federation/externalauthprovider HTTP/1.1 2 Host: login.microsoftonline.com 3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86 64; rv:136.0) Gecko/20100101 Firefox/136.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8 5 Accept-Language: en-US, en; q=0.5 6 Accept-Encoding: gzip, deflate, br 7 Content-Type: application/x-www-form-urlencoded 8 Content-Length: 1979 9 Origin: https://eu-west.azureauth.duosecurity.com 10 Referer: https://eu-west.azureauth.duosecurity.com/ 11 Upgrade-Insecure-Requests: 1 12 Sec-Fetch-Dest: document 13 Sec-Fetch-Mode: navigate 14 Sec-Fetch-Site: cross-site 15 Priority: u=0, i 16 Te: trailers 17 Connection: keep-alive 18 19 token type=Bearer&id token=

eyJhbGciOiJSUzIINiIsImtpZCI6IjZvRVFhZy1penIzRFV1WjNGd3JjcTFTMVhiay1ta054UkdYMm1JcDVraU0iLCJ0eXAiOiJKV1QifQ. eyJpc3MiOiJodHRwczovL2V1LXdlc3QuYXp1cmVhdXRoLmR1b3NlY3VyaXR5LmNvbSIsInN1YiI6Im9nSkdxRmhQQ0ZmMzBveDFSdlZOZV9 ST2LXV19rNzI1MV1KUGVhVG9qTzQiLCJhdWQiOiJZWEJwTFdFeV1UTTBOREZrTG1SMWIzTmxZM1Z5YVhSNUxtTnZiVHBFU1ROV1dFYzVNMH hDUXpSWFNrODBPRXhPVVE9PSIsImV4cCI6MTc0OTcyNzg1OSwiaWF0IjoxNzQ5NzI3NTU5LCJub25jZSI6IkF3QUJFZ0VBQUFBREFPe19CU UQwXzdubWRQdWpKd1MwbmhGSEIyS1JHRnVpYm5TMmVyTXRaQn10LXVCWGF1bVg3Xzd0YWExTE1MX3BCZWNfSHI3U0c1WTBBNm9vYnZvT21P N2kyczBLUnJrZ0FBIiwiRHVvTWZhIjoiTWZhRG9uZSISImFjciI6InBvc3Nlc3Npb25vcm1uaGVyZW5jZSIsImFtciI6WyJwb3AiXX0.dgQ Rx6NhnRFf0RFQqVxzRx0kksTrjrRnHESPFbGrQIuSS-aRGhsUl\_v0fNA-Vg0CVIbtkijQT\_uL5r0APXubNDARY1hW0xd7taIiqZxVgFV-97 FpJc-g10ls38dKSCUnIsNJ4zI50Ual\_352n3BjsDe0fHxSn8FnGN\_gh-ugcXkMdq31E1BmG\_CG80u1yRInnbNnpol3DZUcsA9ktZ2EWemJs gZsVjklZbxUH9xs1SZdNWFWFVBxy7BKopGuT3wYJ2c002GMS8j6QlTIKotqZg1TgMCRySbuT1gatR-LWFII9-HiFcm7ZINiP9yzYpz9WQob 2h-1X21Xw0DvvbbQBkCH6X8rAdLCXIwEtcVEYfdH3Qyh\_RcvpK-ncSA4ByYrF08oTAy5eRxHHh9mgKS1Tg8ZqPLV3gNImNyQ22v\_aZQfqmT 2pAnuFQiSv3bjVeNRdgVSwDiKHsPcpyFCQ\_\_fEmSyrEW-txWicdxbiFgurU2ybM3qEqtMrjnYPFf6v4bcyFWsy-P9TX101tbvQ580vYMr6S Nf0NwtgLUu7aOwheN4ByNNH-Zla9-.Tfe2swGDr8t0-nGo7GIspuBeAXarfVJMu53pQZm584T1hTU4cC0d0BbikIC5uu0ma-Epmh531Cq91 gfCj4tqaopN41mYL3ZNejwbgUH-1J3dhw285B8&cxpires\_in=300&state=

## ID token signed by EAM IDP

HEADER: ALGORITHM & TOKEN TYPE

"alg": "RS256",

"kid": "6oEQag-izr3DUuZ3Fwrcq1S1Xbk-mkNxRGX2mIp5kiM",
"typ": "JWT"

}

PAYLOAD: DATA

"iss": "https://eu-west.azureauth.duosecurity.com",

"sub": "ogJGqFhPCFt30ox1RvVNe\_R0iWW\_k7251YJPeaToj04",
"aud":

"YXBpLWEyYTM0NDFkLmR1b3NlY3VyaXR5LmNvbTpESTNVWEc5M0xCQzR XSk800ExOUQ==",

"exp": 1749727859,

"iat": 1749727559,

"nonce":

"AwABEgEAAAADAOz\_BQD0\_7nmdPujJwS0nhFHB2JRGFuibnS2erMtZBy N-

uBXaemX7\_7taa1LIL\_pBec\_Hr7SG5Y0A6oobvo0m07i2s0KRrkgAA",

"DuoMfa": "MfaDone",

"acr": "possessionorinherence",

"amr": [

"pop'

## Creating our own EAM

- Need to implement this protocol in roadoidc
- Instead of checking MFA, immediately redirect back to Entra ID with signed ID token, using roadoidc signing cert
- Does require roadoidc deployment to Azure App Services since this is no longer just based on static flows.

Oct 16 15:27 •		📥 🌒 🕛 👻
roadtx interactiveauth -u eam@iminyour.cloud -url https://portal.azure.com		
G     Sign in to Microsoft Azure× +	- • 😣	ools/roadoidc/temp ×
(ROADtoo) ← → ×	ම එ ≡	
Performing a TLS handshake to aadcdn.msauth.net		

### One last hurdle

- We need to have an app in our tenant for the EAM:
  - Must have IdP authorize URL as redirect URL registered
  - Must have openid and profile permissions granted
- Solution:
  - Create app in the tenant and grant consent (requires privileged role)
  - Use existing app with these permissions and replace redirect URL on the fly client side.

## Arbitrary MFA for all with EAM

- With roadoidc we can perform fake MFA for any user in scope of the authentication method.
- If we modify the Authentication Methods Policies we can comply with MFA for anyone in scope.
- Does require Global Admin or the MS Graph permission *Policy.ReadWrite.AuthenticationMethod* to configure, so more of a post-exploitation technique.
- EAM does not yet support Auth Strength (even though auth strength is indicated in the protocol)

## Add EAM method for victims

Home > iminyourcloud   Security > Security   Authentication methods > Authentication methods   Policies >				
Yolo MFA External Authentication Methods (Preview)				
Enable and target Configure				
Enable	On			
Include Exclude				
$+$ Add Target $\checkmark$				
Name	Туре	Registration		
All Users	Group	Optional 🗸		

## What if tenant is using EAM?

- If the tenant is using EAM legitimately, we could bypass MFA AD FS style if we can obtain the signing cert and key from the EAM IdP.
- Dumping the cert + key from a third party IdP maybe a bit far fetched.
- What if we can add our own keys?



### Problems with this backdoor

- We can fake MFA but that would affect the security of the victim tenant.
- Could redirect to the real EAM IdP but then issuer would not match.
- User is redirected to untrusted domain.



## Backdoor 2 – attacker flow



## Potential problems with this backdoor

- This is never going to work.
- Why would it be allowed to host the discovery document on a different domain than the issuer.
- Why would it be allowed to host the keys on a different domain than the issuer.

### Actual problem with this backdoor

- It does work
- Because Microsoft

## In practice

$\Box \leftarrow \rightarrow C \qquad \bigcirc \blacksquare$	roadoidcapp.azurewebsites.net/duo/.well-known/openid-configuration		
JSON Raw Data Headers			
Save Copy Collapse All Expand All 🗑 Filter JS	ON		
authorization_endpoint:	"https://eu-west.azureauth.duosecurity.com/authorization"		
<pre>&gt; claims_supported:</pre>	(20)[ "sub", "iss", "cloud_instance_name", "cloud_instance_host_name", "cloud_graph_host_name", " "auth_time", … ]		
<pre>device_authorization_endpoint:</pre>	"https://login.microsoftonline.com/common/oauth2/v2.0/devicecode"		
<pre>end_session_endpoint:</pre>	"https://login.microsoftonline.com/common/oauth2/v2.0/logout"		
<pre>frontchannel_logout_supported:</pre>	true		
<pre>http_logout_supported:</pre>	true		
<pre>vid_token_signing_alg_values_supported:</pre>			
0:	"RS256"		
issuer:	" <u>https://eu-west.azureauth.duosecurity.com</u> "		
jwks_uri:	" <u>https://roadoidcapp.azurewebsites.net/duo/keys</u> "		
<pre>request_uri_parameter_supported:</pre>	false		

$\square  \leftarrow  \rightarrow$	C C A roadoidcapp.azurewebsites.net/duo/keys ■ ☆
JSON Raw Data	Headers
Save Copy Collapse	All Expand All 🖓 Filter JSON
▼ keys:	
▼ 0:	
alg:	"RS256"
e:	"AQAB"
kid:	"6oEQag-izr3DUuZ3Fwrcq1S1Xbk-mkNxRGX2mIp5kiM"
kty:	"RSA"
▼ n:	"vOBcyIKrNcAgnZDMR3hw86uVm_Ryb67eciFvjiwWjSirBABCkX_P3Zni9SV5goViaCj0azjf-EJ6TxiqtIidD-Mjd7o74fZnoqS9WUVme49nXBa nm08Uyg3JBYt4WfuR0k4xaXRRufs3X0iSbQzv3QNrRl3nZ4wfTyzyyJv1z1U0ECk0uD_k3jsd7r21MJ6Wsd7v1A8o4ssoiAi192-QhC8o5_VXgaR
use:	"sig"
▼ x5c:	
• 0:	"MIIFbTCCA1UCFDj0ow9VLk5qr5n2T8Ht8Zq5UHheMA0GCSqGSIb3DQEBCwUAMHMxCZAJBgNVBAYTA1VTMREwDwYDVQQIDAhNaWNoaWdhbjESMBA0 dmeL1JXmChWJoKPRrON/4QnpPGKq0iJ0P4yN3ujvh9meipL1ZRWZ7j2dcFrZCTB2GN/w7UQPUmuLh97FTNhCEKMsF/OLSLvsUU/UqiJMlcUjcEbcx XPVQ4QKTS4P+TeOx3uvaUwnpax3u/UDyjiyyiICLX3b5CELyjn9VeBpH8qG+s1AOyrPiWrBW4jaTTNPR1iOVZUhP33QnaNTzRZUe1VH0+pWYXr2P2 vtkC76EyLeqS8LdaeCbYh3c6h7Cdudp8wxkTb5+9eUsP87PVkeNOv0bMQz3JR2DkXCgVr3uSwdpEPcNiSjWEUGQwFIQONItWW9JVmmKSX1IBKeLC2
▼ x5t#S256:	
0:	"6oEQag-izr3DUuZ3Fwrcq1S1Xbk-mkNxRGX2mIp5kiM"
▼ 1:	
alg:	"RS256"
e:	"AQAB"
kid:	"ROIDKEY"
kty:	"RSA"
▼ n:	"1IaWDe8HR+3fRWi6oinVAg8mW4NY9+Rm7Rv3/+2yw1Uz8ThNFmXdRYFXUitCixm8jiBsVB0kxm6MrgVyeUxjgoq4ZtNtEfapxhtwG/jw8+CCDcp jzAojy9JVEQM8AUWPgGEWHiONVXNw6wrepB9c2H6K+hd/7gZhiYjUg0yIj9k2Ny089tf1gMzGFiY1Ho57JmWdXZDqG1+beAugTxe1RMKoieH+mf6,
use:	"sig"
▼ x5c:	
♥ 0:	"MIIFGTCCAwGgAwIBAgIUYN8XVMv+W33sWVMv9NKfITbLydEwDQYJKoZIhvcNAQELBQAwHDEaMBgGA1UEAwwRYXBwQ2VydEZvckF6dXJlQUQwHhc CnSP809BapeYTHKSSBHNxQmkUC+TES1KzdOU3Vd8YJlep3aHEwa6188vPrSrBi2OD1BKihsVWhHQCKSqwPgKBNKS03mdjKGe9QyueW9eXmotfzUI ksYCOD1H4MINvSqcX7o7qfTT2dWpSGdFdahNceo0bGgq3HEAdAgMBAAGjUzBRMB0GA1UdDgQWBBQ1Jxtsr6pZESAEEc79XJtKaVFHojAfBgNVHSM SDHwk+VIkgdurNzkbH3AhO2G2JAqizDo3KOmDhsZahMqLwEjakK3kTvcZk1Hw3Ls5QDhJkiG2EkUGeVtVx+laZc1yYNg75Le4+ZTsu0GeDc59Sp7

#### Demo

0

5

4

	â				
	user@ubuntu:~/I	ROADtools		🛛	
	user@ubuntu:~/RC (ROADtools) → ROADtools git:(master) × roadtx -p 127.0.0.1:8082 keepassauth -u d	DADtools 155x43 Juo2@iminyour.cloud			
					Cattlear
1					Settings
				reques	ts and responses passing throug
					Replace
				anspor	(-Security.*\$
				15 *	X-X33-Protection: 0
				KU.*	
				tps://eu	u-west.azur action="https://road
10				ocket n	nessages passing through the Pro
	the Configuration library		an a literate		
		Uniy apply to in-sci	opeitems		
		Add Enabled	Direction	Match	Replace
		Edit			

## Recap

- Discovery URL we configure:
  - <u>https://roadoidcapp.azurewebsites.net/duo/.well-known/openid-configuration</u>
- Discovery document gives issuer (for DUO):
  - <u>https://eu-west.azureauth.duosecurity.com</u>
- Discovery document keys URL:
  - <u>https://roadoidcapp.azurewebsites.net/duo/keys</u>
  - Gives both the backdoor keys + the real keys fetched from DUO
- Discovery document gives the real authorization page from DUO
  - Real users get redirected to DUO and MFA keeps working as it should
  - Attacker can intercept the redirect and send it to roadoidc to bypass DUO
  - Only Entra ID communicates with roadoidc (so no domain in EDR/FW logs)

## Let's read the specs

#### 4.3. OpenID Provider Configuration Validation

If any of the validation procedures defined in this specification fail, any operations requiring the information that failed to correctly validate MUST be aborted and the information that failed to validate MUST NOT be used.

The issuer value returned MUST be identical to the Issuer URL that was used as the prefix to /.well-known/openidconfiguration to retrieve the configuration information. This MUST also be identical to the iss Claim value in ID Tokens issued from this Issuer.

Ref: https://openid.net/specs/openid-connect-discovery-1\_0.html#ProviderConfigurationValidation

тос

### Let's read Microsoft's own docs

#### Discovery of provider metadata

An external identity provider needs to provide an OIDC Discovery endpoint 2<sup>o</sup>. This endpoint is used to get more configuration data. The *full* URL, including *.well-known/oidc-configuration*, must be included in the Discovery URL configured when the EAM is created.

The endpoint returns a Provider Metadata JSON document <sup>27</sup> hosted there. The endpoint must also return the valid content-length header.

The following table lists the data that should be present in the metadata of the provider. These values are required for this extensibility scenario. The JSON metadata document may contain more information.

For the OIDC document with the values for Provider Metadata, see Provider Metadata 2.

#### C Expand table

Metadata value	Value	Comments
lssuer		This URL should match both the host URL used for discovery and the iss claim in the tokens issued by the provider's service.

## Real EAM or not?

Home > iminyourcloud   Security >	Security   Authentication methods > Authentication methods   Policies >
<b>Cisco Duo 2</b> External Authentication Methods (Preview)	
🔟 Delete	
Enable and target <b>Configure</b>	
Method Properties	
Your provider will give you the name, clie	ent ID, discovery endpoint, and app ID for the external authentication method.
Name *	Cisco Duo 2
	The provider name cannot be changed.
Client ID *	YXBpLWEyYTM0NDFkLmR1b3NIY3VyaXR5LmNvbTpESTNVWEc5M0xCQzRXSk8
Discovery Endpoint *	https://eu-west.azureauth-duosecurity.com/.well-known/openid-configuration
App ID *	0d187fb5-d1d7-45e9-b567-b270338ad8a5
Request admin consent	Admin consent granted



# CA Custom Controls

## Custom controls in Conditional Access

- Essentially the predecessor of EAM
- Been around for a few year, announced in 2020 that it would not become GA
- Companies still use it
- Until EAM was around the only way to use external MFA providers

## Custom controls

Home > iminyourcloud   Security > Security   Conditional Access > Conditional Acc	Grant	>
DUO …		
Conditional Access policy	Block access	
🔟 Delete 💿 View policy information 💿 View policy impact	• Grant access	
Control access based on Conditional Access policy to bring signals together, to make	Require multifactor ① authentication	
decisions, and enforce organizational policies. Learn more 🗗	Require authentication () strength	
Name *	Require device to be	)
DUO	marked as compliant	
Assignments	Require Microsoft Entra (i hybrid joined device	)
Users (i)	Require approved client	
Specific users included	app See list of approved client apps	
Target resources ()		
All resources (formerly 'All cloud apps')	policy See list of policy protected client	:
Network NEW ①	apps	
Not configured	Require password change 🛈	
Conditions ①	🗸 RequireDuoMfa	
0 conditions selected	RequireDuoMfaa	
Enable policy	Require valid client cert (Preview	/)
Report-only On Off		
Save	Select	

## Custom control configuration json

```
\mathcal{P} Search resources, services, and docs (G+/)
     Microsoft Azure
                                                                                       🧳 Copilot
RequireDuoMfa
                                                                                                               \times
Enter the JSON for customized controls given by your claim providers.
      "Name": "Duo Security",
      "AppId": "768d6943-5512-40ae-9da2-49acc9fa8e80",
      "ClientId": "YXBpLWEyYTM0NDFkLmR1b3N1Y3VyaXR5LmNvbTpESVBM0VpTOTJONFU4Q0s0MzRCUA==",
      "DiscoveryUrl": "https://eu-west.azureauth.duosecurity.com/.well-known/openid-configuration"
      "Controls": [
          "Id": "RequireDuoMfa",
          "Name": "RequireDuoMfa",
          "ClaimsRequested": [
              "Type": "DuoMfa",
              "Value": "MfaDone",
              "Values": null
          "Claims": null
```



### Custom controls and OIDC

- Implementation is identical between EAM and Custom Controls.
- Can perform the exact same attacks: modify the Custom Control discovery URL and then inject backdoor keys.
- No need for app registration and redirect URL check in this case.
- Slightly different response is required to make it work.

# Disclosure process

### Disclosure process

- Sent two reports to MSRC
  - One describing the bug in Custom Controls
  - One describing the bug in EAM
- The bug in EAM was closed as a duplicate of the Custom Controls report
- The bug in Custom Controls was closed as "not a vulnerability" since "admins are free to change the URL at any time and you need privileged access to do this".
- They also pointed out that Custom Controls will be replaced by EAM which is "better".

## What could be improved

- Implement the OAuth mandatory security checks
- Don't use the discovery URL pattern but use the issuer and then do the discovery based on the *.well-known/openid-configuration* suffix.
  - This is what is used in federated credentials on apps, which is why this attack doesn't work there.

# Attack flow and detection

## Modifying or adding EAM

- Can be done in the Azure portal / Entra Admin portal
- Both will use the Microsoft Graph endpoint /authenticationMethodsPolicy/authenticationMethodConfigurations/
- Quick detection for policy modifications (assuming you have Graph Activity logs)

MicrosoftGraphActivityLogs | where RequestUri contains "authenticationMethodsPolicy" | where RequestMethod == "PATCH"

 Also recorded in the Entra audit logs, where we can get the actual changes

## Detection: EAM modification

/	TargetResources		ources [{"id":null,"displayName":			displayName":null,"modifiedProperties":[{"displayName":"Authent
	$\sim$	0	{"io	l":null,"di	splayName	:null,"modifiedProperties":[{"displayName":"AuthenticationMethodsPolicy","oldValue":"\"{\\\"id\\\":\\\"authenticationMethodsPolicy\\\",\\\"display
			admi	nistrative	Units	]
			displ	ayName		null
			id	-		null
		~	mod	ifiedProp	erties	{"displayName":"AuthenticationMethodsPolicy","oldValue":"\"{\\\"id\\\":\\\"authenticationMethodsPolicy\\\",\\\"displayName\\\":\\\"Authenticati
			$\sim$	0	{"displayNa	me":"AuthenticationMethodsPolicy","oldValue":"\"{\\\"id\\\":\\\"authenticationMethodsPolicy\\\",\\\"displayName\\\":\\\"Authentication Methods
				d	splayName	AuthenticationMethodsPolicy
				n	ewValue	"{\"id\":\"authenticationMethodsPolicy\",\"displayName\":\"Authentication Methods Policy\",\"description\":\"The tenant-wide policy that controls ngs\":{\"includeTarget\":{\"id\":\"all_users\",\"targetType\":1},\"state\":0,\"voiceReportingCode\":0},\"microsoftAuthenticatorPlatformSettings\":{\"en \"ExcludeTarget\":{\"TargetType\":1,\"Id\":\"00000000-0000-0000-00000000000\"}},\"numberMatchingRequiredState\":{\"State\":0,\"IncludeTarget\":{\"IncludeTarget\":{\"TargetType\":1,\"Id\":\"00000000-0000-0000-0000000000\"},\"numberMatchingRequiredState\":{\"State\":0,\"IncludeTarget\":{\"IncludeTarget\":{\"TargetType\":1,\"Id\":\"00000000-0000-0000-00000000000\"},\"numberMatchingRequiredState\":{\"State\":0,\"IncludeTarget\":{\"IncludeTarget\\":{\"IncludeTarget\":{\"IncludeTarget\":{\"IncludeTarget\":{\"IncludeTarget\":{\"IncludeTarget\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"IncludeTarget\\":{\"Include
				o	dValue	"{\"id\":\"authenticationMethodsPolicy\",\"displayName\":\"Authentication Methods Policy\",\"description\":\"The tenant-wide policy that controls ngs\":{\"includeTarget\":{\"id\":\"all_users\",\"targetType\":1},\"state\":0,\"voiceReportingCode\":0},\"microsoftAuthenticatorPlatformSettings\":{\"en \"ExcludeTarget\":{\"TargetType\":1,\"ld\":\"00000000-0000-0000-000000000000\"}},\"numberMatchingRequiredState\":{\"State\":0,\"Include \"ID\":\"all_users\",\"IsRegistrationRequired\":false}],\"defaultLifetimeInMinutes\":120,\"defaultLength\":8,\"minimumLifetimeInMinutes\":120,\"max \":1,\"usableFor\":0,\"excludeTargets\":[]},{\"includeTargets\":[{\"TargetType\":1,\"ID\":\"all_users\",\"IsRegistrationRequired\":true}],\"id\":\"Passworc ertificateExtensionFilters\":{\"ekuOIDs\":[]},\"certificateAuthorityScopes\":[],\"crIValidationConfiguration\":{\"State\":0,\"ExemptedCertificateAuthor NDFkLmR1b3NIY3VyaXR5LmNvbTpESTNVWEc5M0xCQzRXSk80OExOUQ==\",\"discoveryUrl\":\"https://roadoidcapp.azurewebsites.net/duo/.well- {\"clientId\":\"YXBpLWEyYTM0NDFkLmR1b3NIY3VyaXR5LmNvbTpESTNVWEc5M0xCQzRXSk80OExOUQ==\",\"discoveryUrl\":\"https://roadoidcapp.acurewebsites./":"https://roadoidcapp./

```
AuditLogs
 1
     where OperationName == "Authentication Methods Policy Update"
 2
      extend modifiedProps = TargetResources[0].modifiedProperties
 3
      extend initiatedUser = tostring(InitiatedBy.user.userPrincipalName)
 4
 5
     mv-expand modifiedProps
      extend actualnewvalue = parse_json(tostring(modifiedProps.newValue))
 6
      extend mfa = parse json(tostring(actualnewvalue)).authenticationMethodConfigurations
 7
     mv-expand mfa
 8
 9
      extend oidc = parse json(tostring(mfa)).openIdConnectSetting
      where isnotnull(oidc)
10
      extend discovery = oidc.discoveryUrl
11
      extend actualoldvalue = parse json(tostring(modifiedProps.oldValue))
12
      extend oldmfa = parse json(tostring(actualoldvalue)).authenticationMethodConfigurations
13
     mv-expand oldmfa
14
      extend oldoidc = parse json(tostring(oldmfa)).openIdConnectSetting
15
      where isnotnull(oldoidc)
16
      where tostring(parse json(tostring(mfa)).id) == tostring(parse json(tostring(oldmfa)).id)
17
      extend olddiscovery = oldoidc.discoveryUrl
18
      where tostring(olddiscovery) !~ tostring(discovery)
19
20
      project initiatedUser, olddiscovery, discovery, mfa.displayName
21
```

#### Results Chart

initi	atedUser	olddiscovery	discovery	mfa_displayName
>	dirkjan@iminyour.cloud	https://roadoidcapp.azurewebsites.net/	https://eu-west.azureauth.duosecurity.com/.well-known/openid	Cisco Duo 2
>	dirkjan@iminyour.cloud	https://roadoidcapp.azurewebsites.net/	https://eu-west.azureauth.duosecurity.com/.well-known/openid	Cisco Duo 2
>	dirkjan@iminyour.cloud	https://roadoidcapp.azurewebsites.net/	https://roadoidcapp2.azurewebsites.net/duo/.well-known/openi	Cisco Duo

## Alternative approach

- Use Azure AD Graph "legacy" API <u>https://graph.windows.net</u>
- Modify the "policy" object over an internal API version
- No useful logging  $\odot$  /  $\otimes$

$\sim$	TargetResources		urces [{"id":"8	[{"id":"8c8fd8dc-b179-480b-90f9-f622e5531d2f","displayName":"Default Poli			
	V 0 {"id":"8c8fd8dc-b179-4 administrativeUnits displayName id > modifiedProperties type		{"id":"8c8fd8dc-b179-	-480b-90f9-f622e5531d2f","displayName":"Default Policy","type":"Policy","modifiedProperties":[			
			administrativeUnits	0			
			displayName	Default Policy			
			id	8c8fd8dc-b179-480b-90f9-f622e5531d2f			
			modifiedProperties	[{"displayName":"Included Updated Properties","oldValue":null,"newValue":"\"\""}]			
			type	Policy			

## Modifying Custom Controls

- Uses <u>https://main.iam.ad.ext.azure.com/api/ClaimProviders</u> endpoint.
- No "public" API to perform modification.
- Used to not generate any useful logging when I reported it
- Is now recorded in the audit log properly
- Modifying Custom Controls / CA policies via internal API was blocked a few years ago

## Custom Control modification

lew Query 1* $\cdots  imes +$		
▶ Run Time range	e : Last 24 hours Show : 1000 results	
1 AuditLogs 2		
Results Chart		
TimeGenerated [UTC] $\uparrow \downarrow$	ResourceId	OperationName
✓ 6/12/2025, 7:44:00.097 AM	/tenants/6287f28f-4f7f-4322-9651-a8697d8fe1bc/provider	Update policy
TenantId	7f331091-129a-43a6-86ce-140a642f2d31	
SourceSystem	Azure AD	
TimeGenerated [UTC]	2025-06-12T07:44:00.0970916Z	
Resourceld	/tenants/6287f28f-4f7f-4322-9651-a8697d8fe1bc/providers/M	licrosoft.aadiam
OperationName	Update policy	
OperationVersion	1.0	
Category	Policy	
ResultSignature	None	
DurationMs	0	
CorrelationId	e69f756f-438c-4ee5-92ab-1b778d894691	
Resource	Microsoft.aadiam	

## Modified DiscoveryUrl

~	mod	lifiedP	rop	erties		[{"c	lisplay	/Name":"PolicyDe	tail","oldValue":"[\"{\\\"Version\\\":0,\\\"LastUpdatedTimestamp\\\":\\\"2025-06-12T07:43:35.8289374Z\\\",\\\"State\\\":\\\
	$\sim$	0	{" <b>displayName":"PolicyDetail","oldValu</b> displayName PolicyDetail			":"Poli	icyDetail","oldValu	ue":"[\"{\\\"Version\\\":0,\\\"LastUpdatedTimestamp\\\":\\\"2025-06-12T07:43:35.8289374Z\\\",\\\"State\\\":\\\"Disabled\\`	
						Pol	icyDetail		
		✓ newValue ["{\"Version\":0,\"Las		\"Version\":0,\"Las	stUpdatedTimestamp\":\"2025-06-12T07:43:59.9766406Z\",\"State\":\"Disabled\",\"ClaimsProviders\":[{\"Name\":\"Duo Sec				
				<ul><li>✓ 0</li></ul>		{" <b>\</b>	/ersior	n":0,"LastUpdated	Timestamp":"2025-06-12T07:43:59.9766406Z","State":"Disabled","ClaimsProviders":[{"Name":"Duo Security","AppId":"768c
			<ul> <li>ClaimsProviders</li> <li>0 {"Name":"Duo</li> </ul>		viders	[{"Name":"Duo Security","AppId":"768d6943-5512-40ae-9da2-49acc9fa8e80","ClientId":"YXBpLWEyYTM0NDFkLmR1b3N			
					{"Name":"Duc	o Security","AppId":"768d6943-5512-40ae-9da2-49acc9fa8e80","ClientId":"YXBpLWEyYTM0NDFkLmR1b3NIY3VyaXR5LmNv			
								AppId	768d6943-5512-40ae-9da2-49acc9fa8e80
								ClientId	YXBpLWEyYTM0NDFkLmR1b3NlY3VyaXR5LmNvbTpESVBMOVpTOTJONFU4Q0s0MzRCUA==
							>	Controls	[{"Id":"RequireDuoMfa","Name":"RequireDuoMfa","ClaimsRequested":[{"Type":"DuoMfa","Value":"MfaDone"}]}]
								DiscoveryUrl	https://eu-west.azureauth.duosecurity.com/.well-known/openid-configuration
								Name	Duo Security

#### 1 AuditLogs

2	<pre>where OperationName == "Update policy"</pre>
3	<pre>extend modifiedProps = TargetResources[0].modifiedProperties</pre>
4	<pre>extend initiatedUser = tostring(InitiatedBy.user.userPrincipalName)</pre>
5	mv-expand modifiedProps
6	<pre>extend newvalue = modifiedProps.newValue</pre>
7	mv-expand newvalue
8	<pre>extend actualnewvalue = parse_json(tostring(newvalue))[0]</pre>
9	<pre>extend claimproviders = parse_json(tostring(actualnewvalue)).ClaimsProviders</pre>
10	mv-expand claimproviders
11	<pre>extend newdiscoveryurl = claimproviders.DiscoveryUrl</pre>
12	<pre>extend oldvalue = modifiedProps.oldValue</pre>
13	mv-expand oldvalue
14	<pre>extend actualoldvalue = parse_json(tostring(oldvalue))[0]</pre>
15	<pre>extend oldclaimproviders = parse_json(tostring(actualoldvalue)).ClaimsProviders</pre>
16	mv-expand oldclaimproviders
17	<pre>extend olddiscoveryurl = oldclaimproviders.DiscoveryUrl</pre>
18	<pre>where tostring(oldclaimproviders.ClientId) == tostring(claimproviders.ClientId)</pre>
19	<pre>where tostring(oldclaimproviders.Name) == tostring(claimproviders.Name)</pre>
20	<pre>where tostring(olddiscoveryurl) !~ tostring(newdiscoveryurl)</pre>
21	<pre>extend clid = oldclaimproviders.ClientId</pre>
22	extend providername = claimproviders.Name

23 | project initiatedUser, providername, olddiscoveryurl, newdiscoveryurl

#### Results Chart

nitiatedUser		providername	olddiscoveryurl	newdiscoveryurl
> dirkja	n@iminyour.cloud	Duo Securitya	https://roidapp.azurewebsites.n	https://roadoidcapp.azurewebsites.net/duo/.well-known/openid-configuration
> dirkja	n@iminyour.cloud	Duo Securitya	https://roadoidcapp.azurewebsi	https://roadoidcapp2.azurewebsites.net/duo/.well-known/openid-configuration

# Conclusions

## Conclusions

- Federated credentials provide new opportunities for taking control and persisting on applications and managed identities – new things to monitor for.
- EAM can be configured as MFA method for a broad scope, helping in post-exploitation scenarios.
- If you actually use EAM or Custom Controls in CA, be on the lookout for "backdoor keys", which only works because Microsoft refuses to actually implement mandatory OAuth2 security checks.
- New roadoidc release will make this feature available soon.



## BYO IDP in Entra ID

Persisting and bypassing MFA with OIDC based protocols