

# Modern Workplace Advanced

**Azure Hands On Lab May 8<sup>th</sup> 2020**

V3.0 by :      Stephan van de Kruis  
                 Gido Weekens



2tCloud



## Content

Introduction .....	3
Activity 1 : Use Secure Score and Passwordless Sign-in to secure your tenant.....	4
Exercise 1a : Setting up the user accounts .....	4
Exercise 1b : Use Secure Score to harden your security .....	7
Exercise 1c: Implement Passwordless Sign-in .....	13
Exercise 1d: Implement custom Conditional Access policy .....	17
Exercise 1e: Validate Passwordless Sign-in .....	20
Activity 2 : Graph API Explorer .....	24
Activity 3 : Connect with the Graph API.....	25
Activity 4 : Custom Graph API commands .....	28
Activity 5 : MSIX.....	31
Exercise 1a: Create Packaging VM.....	31
Exercise 1b: Test your package .....	34
Deploy the MSIX application with Intune .....	34
Test the Package locally .....	36
Extra resources .....	39

## Introduction

*Estimated time to complete this lab*

120 minutes

### *Objectives*

During this lab, you will learn how to get started with Azure to;

- 

### *Prerequisites*

To complete this course, you will be needing;

- Laptop/computer with Internet browser and Wi-Fi connected
- Account with an Azure CSP Subscription
- A Microsoft 365 business subscription

### *Materials*

All student materials are available for download here:

<https://github.com/Copaco/handsonlab/>



## Activity 1 : Use Secure Score and Passwordless Sign-in to secure your tenant

*Estimated time to complete this activity*

45 minutes

### Objectives

In this activity, you will configure the components necessary to perform this lab;

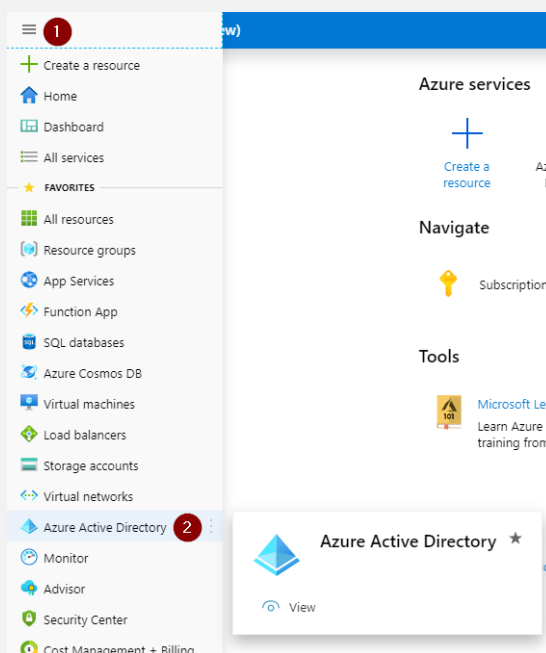
- Use Microsoft Secure Score to secure your Microsoft 365 tenant
- Enable baseline policies for MFA
- Enable passwordless authentication for users
- Troubleshoot Conditional Access

### Exercise 1a : Setting up the user accounts

- 1) Using your *Work Account*, you can sign into the Azure Portal using:

<https://portal.azure.com>

- 2) Browse to the Azure Active Directory

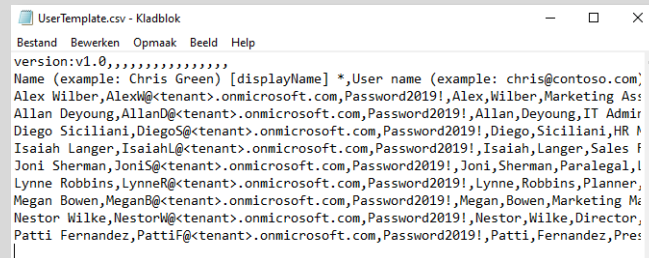


- 3) Note the tenant name from the Azure Active Directory blade

<tenant>.onmicrosoft.com

- 4) From GitHub, [download](#) **UserTemplate.csv** and [save](#) it to your local computer.

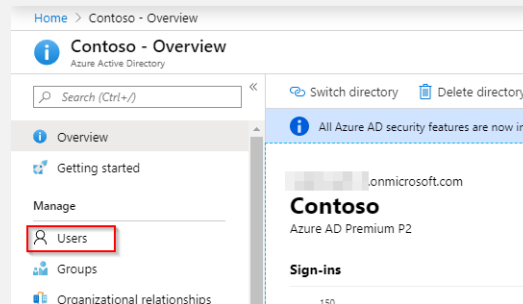
- 5) [Edit](#) the CSV-file using Notepad. Find and replace <tenant> so it corresponds with the tenant you're using. **Save** your changes.



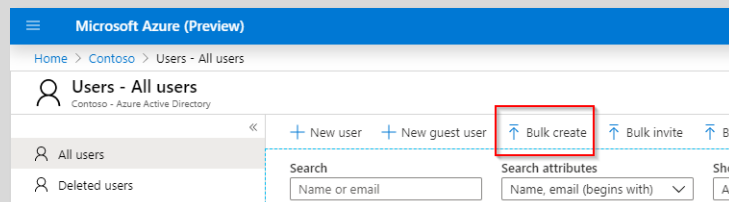
```

UserTemplate.csv - Kladblok
Bestand Bewerken Opmaak Beeld Help
version: v1.0,,,,,,,,,,,,,
Name (example: Chris Green) [displayName] *,User name (example: chris@contoso.com),
Alex Wilber,AlexW@<tenant>.onmicrosoft.com,Password2019!,Alex,Wilber,Marketing As
Allan Deyoung,AllanD@<tenant>.onmicrosoft.com,Password2019!,Allan,Deyoung,IT Admir
Diego Siciliani,DiegoS@<tenant>.onmicrosoft.com,Password2019!,Diego,Siciliani,HR
Isaiah Langer,IsaiahL@<tenant>.onmicrosoft.com,Password2019!,Isaiah,Langer,Sales f
Joni Sherman,JoniS@<tenant>.onmicrosoft.com,Password2019!,Joni,Sherman,Paralegal,l
Lynne Robbins,LynneR@<tenant>.onmicrosoft.com,Password2019!,Lynne,Robbins,Planner
Megan Bowen,MeganB@<tenant>.onmicrosoft.com,Password2019!,Megan,Bowen,Marketing M
Nestor Wilke,NestorW@<tenant>.onmicrosoft.com,Password2019!,Nestor,Wilke,Director
Patti Fernandez,PattiF@<tenant>.onmicrosoft.com,Password2019!,Patti,Fernandez,Pres
  
```

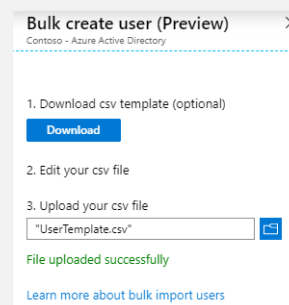
- 6) Open the [Users](#) blade



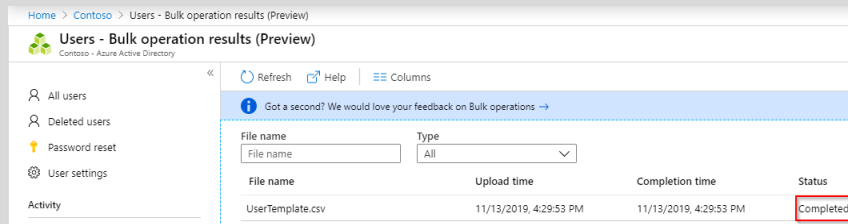
- 7) Choose [Bulk create](#)



- 8) Upload the edited CSV and Submit.



- 9) Click to watch the status of the import job. Make sure it's successful before you proceed. Refresh the blade for status update,



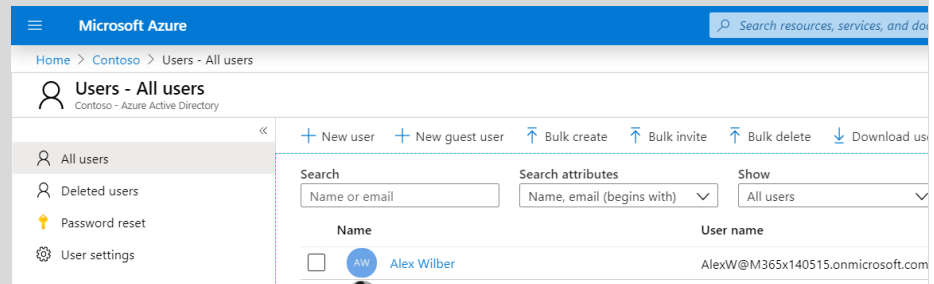
- 10) Open a new InPrivate browser window

- 11) Browse to Office.com and sign in with user Alex Wilber. Take note of the current time.

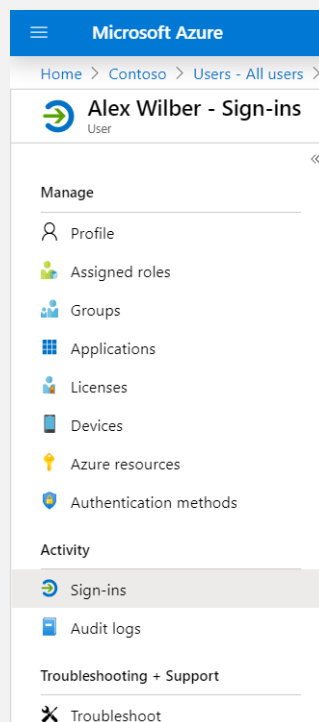
Username *AlexW@<tenant>.onmicrosoft.com* &  
Password *Password2019!*

- 12) Switch back to the regular browser with the opened Azure Portal

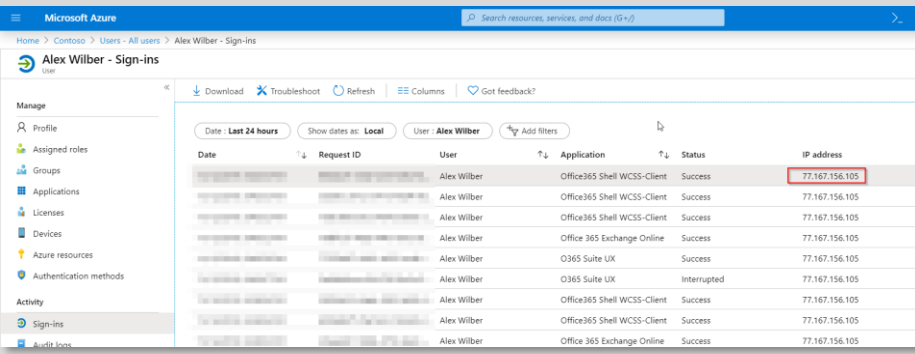
- 13) Browse to the Azure Active Directory and then Users



- 14) Open Alex Wilber and open Sign-ins from the navigation bar.



- 15) Select the most recent sign-in, which corresponds with the sign-in performed in step 11. There's some latency, so wait and refresh if the sign-in won't show.



Date	Request ID	User	Application	Status	IP address
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office365 Shell WCCS-Client	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office365 Shell WCCS-Client	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office365 Shell WCCS-Client	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office 365 Exchange Online	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	O365 Suite UX	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	O365 Suite UX	Interrupted	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office365 Shell WCCS-Client	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office365 Shell WCCS-Client	Success	77.167.156.105
7/16/2020 10:10:10 AM	80000000-0000-0000-0000-000000000000	Alex Wilber	Office 365 Exchange Online	Success	77.167.156.105

- 16) Take note of the IP-address that was used to perform the sign-in. You will need this later.

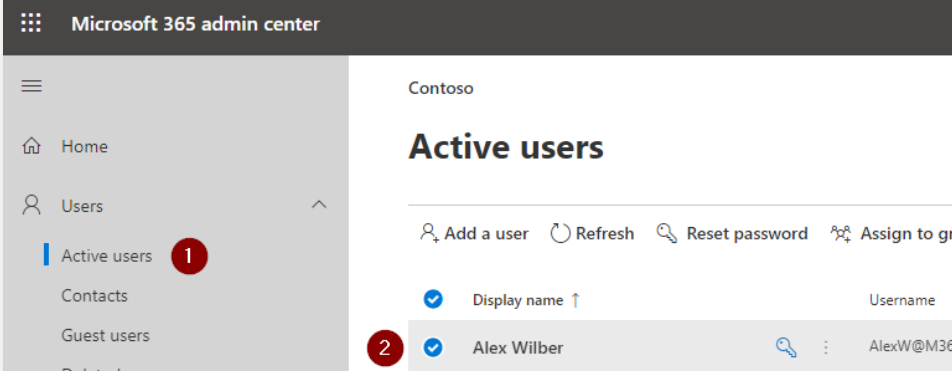
- 17) You're done for this exercise!

## Exercise 1b : Use Secure Score to harden your security

- 1) Using your *Work Account*, you can sign into the Azure Portal using:

<https://admin.microsoft.com>

- 2) Browse to Users. Select user Alex from the list and assign the Microsoft 365 license to the user.



Microsoft 365 admin center

Contoso

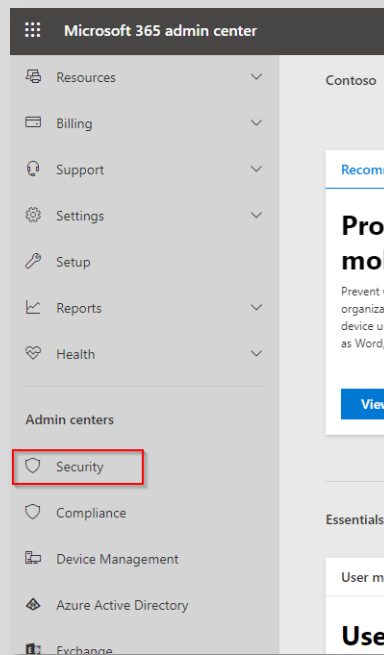
### Active users

[Add a user](#)
[Refresh](#)
[Reset password](#)
[Assign to group](#)

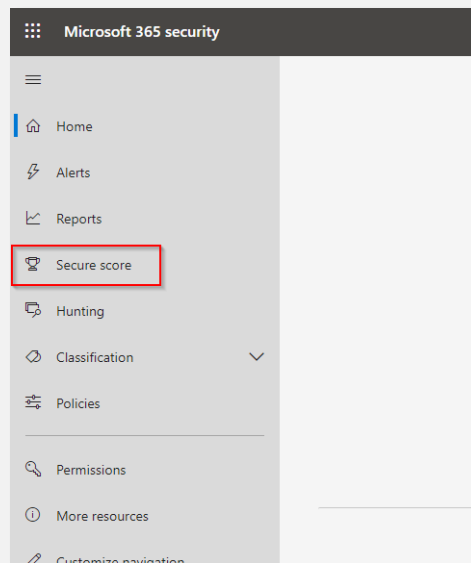
Display name ↑	Username
Alex Wilber	AlexW@M365



- 3) From the Microsoft 365 Admin Center navigation panel, browse to the [Security Center](#).

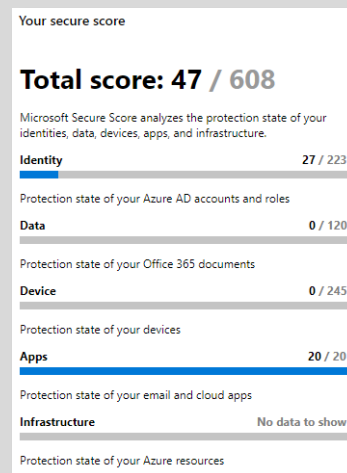


- 4) Open [Secure Score](#) from the navigation panel.





5) Review the score overview.



6) Open the *Improvement Actions* tab.

**Microsoft Secure Score**

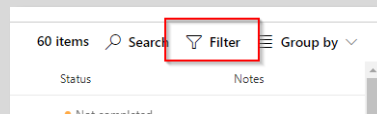
Overview **Improvement actions** History

Actions you can take to improve your Microsoft Secure Score. Score updates may take up to 24 hours.

Export

Improvement action	Rank	Score	Category	User impact	Implementation cost
Require MFA for Azure AD privileged roles	1	0/50	Identity	Low	Low
Require MFA for all users	2	0/20	Identity	Moderate	Moderate

7) Click the *Filter* icon.



8) Make sure you only select the items with minimal *User impact* and minimal *Implementation Cost*. Click **Apply**.

**Filters** Clear filters

**Category**

- ☐ Identity
- ☐ Data
- ☐ Device
- ☐ Apps
- ☐ Infrastructure

**User impact**

- ☐ High
- ☐ Moderate
- ☒ Low

**Implementation cost**

- ☐ High
- ☐ Moderate
- ☒ Low

- 9) Find and select  
*Turn on audit data recording*

## Microsoft Secure Score

Overview Improvement actions History

Actions you can take to improve your Microsoft Secure Score. Score updates may take up to 24 hours.

↓ Export

Applied filters: User impact: Low × Implementation cost: Low ×

Improvement action	Rank	Score	Category	User impact	Implementation cost
Require MFA for Azure AD privileged roles	1	0/50	Identity	Low	Low
<input checked="" type="checkbox"/> Turn on audit data recording	3	0/15	Data	Low	Low
Set outbound spam notifications [Not Scored]	7	0/15	Data	Low	Low

- 10) Review the  
*Description* and  
*Next steps* from the  
side panel. **Review**  
the setting. The  
*Security & Compliance Center*  
will be opened.

## Turn on audit data recording

0/15 points

### Status

● Not completed

### Description

Turning on audit data recording for your Office 365 service ensures that you have a record of every user and administrator's interaction with the service, including Azure AD, Exchange Online, and SharePoint Online/OneDrive for Business. This data makes it possible to investigate and scope a security breach, should it ever occur. All activity is recorded and retained for 90 days.

Feature in place: false.

### Category

Data

### User impact

Low

### Protects against

Account Breach  
Data Exfiltration  
Data Deletion  
Elevation of Privilege  
Malicious Insider

### Complexity

Low

### Next steps

On the Audit log search page, turn on audit logging by selecting **Start recording user and admin activity**. If you don't see this link, audit log search has already been turned on and you can start investigating.

### How will this affect my users?

This change has no known impact on your users.

### Compliance Controls

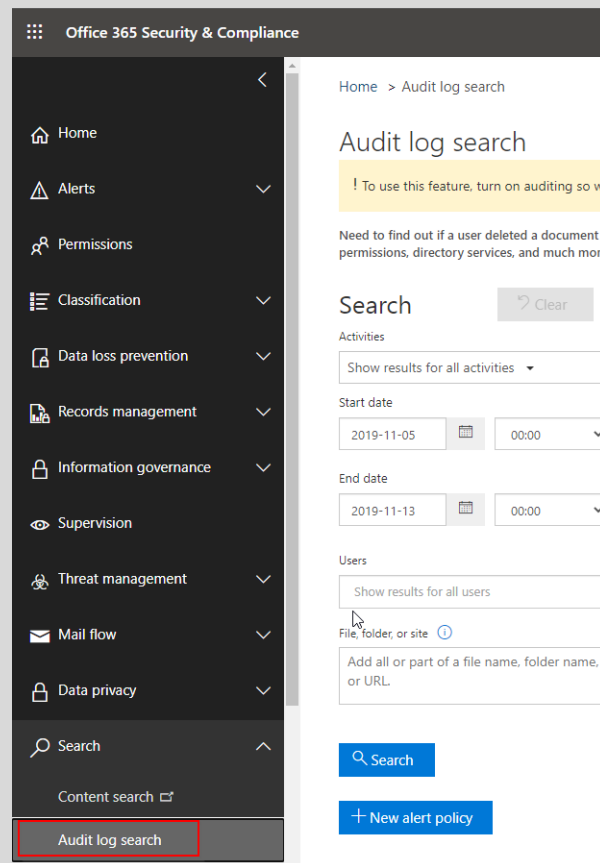
CSA CCM301: Control IAM-01  
FedRAMP Moderate: Control AU-9  
GDPR: Control 6.9.4  
ISO 27001:2013: Control A.12.4.2  
ISO 27018:2014\_ID: Control C.12.4.2, Part 1  
NIST 800-171: Control 3.3.8  
NIST 800-53: Control AU-9

**Review**

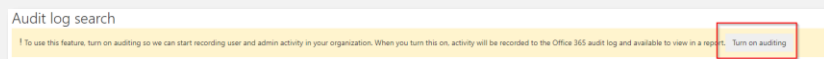
Save

Ignore

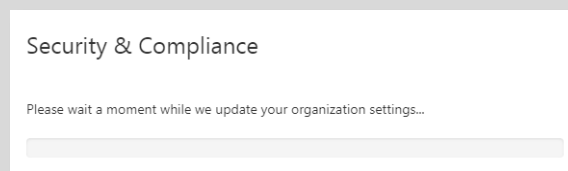
11) Browse to the [Audit log search](#).



12) If not enabled already, **Turn on auditing**.



13) Wait for the change to be effective. This could take a few minutes. Close the browser tab with the [Compliance and Security Center](#).

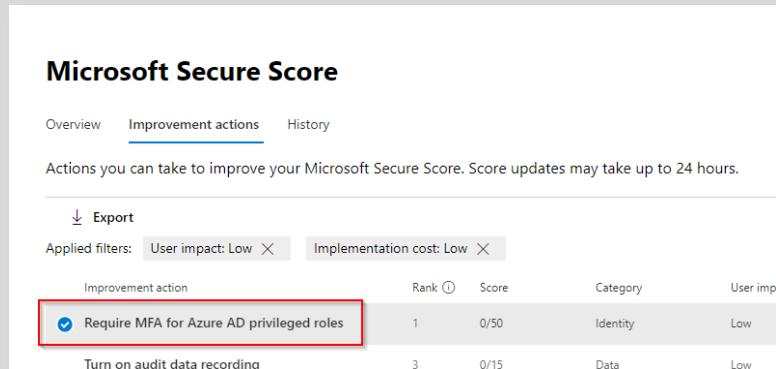


14) You can check Secure Score to see an updates score. However, this usually takes 24 hours to show. Come back later to see how your score has changed from the actions taken. Proceed with the lab for now.

You're done with  
this exercise!

## Exercise 1c: Implement Passwordless Sign-in

- 1) From the [Secure Score Improvement Actions](#), select the [Require MFA for Azure AD privileged roles](#).



**Microsoft Secure Score**

Overview **Improvement actions** History

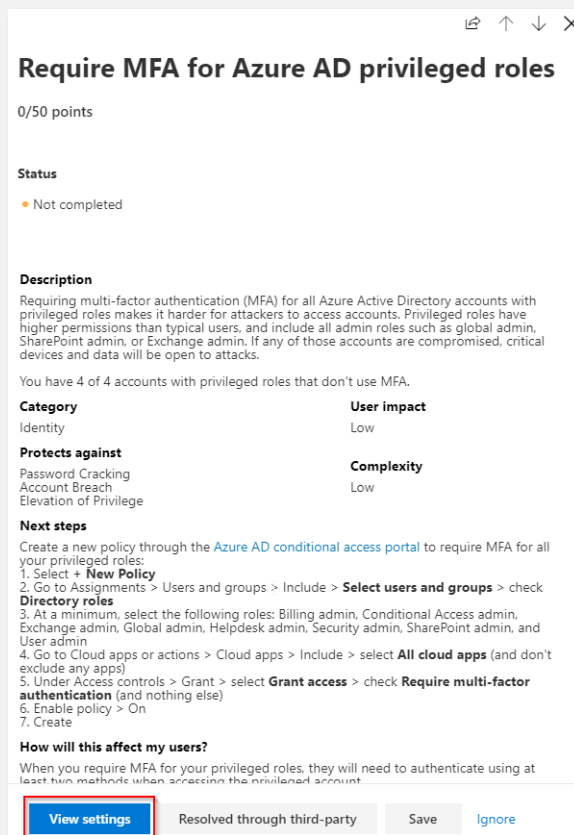
Actions you can take to improve your Microsoft Secure Score. Score updates may take up to 24 hours.

Export

Applied filters: User impact: Low X Implementation cost: Low X

Improvement action	Rank	Score	Category	User impact
<input checked="" type="checkbox"/> <b>Require MFA for Azure AD privileged roles</b>	1	0/50	Identity	Low
Turn on audit data recording	3	0/15	Data	Low

- 2) Review the details and click the **View settings** button.



**Require MFA for Azure AD privileged roles**

0/50 points

**Status**

Not completed

**Description**

Requiring multi-factor authentication (MFA) for all Azure Active Directory accounts with privileged roles makes it harder for attackers to access accounts. Privileged roles have higher permissions than typical users, and include all admin roles such as global admin, SharePoint admin, or Exchange admin. If any of those accounts are compromised, critical devices and data will be open to attacks.

You have 4 of 4 accounts with privileged roles that don't use MFA.

Category	User impact
Identity	Low

Protects against	Complexity
Password Cracking Account Breach Elevation of Privilege	Low

**Next steps**

Create a new policy through the [Azure AD conditional access portal](#) to require MFA for all your privileged roles:

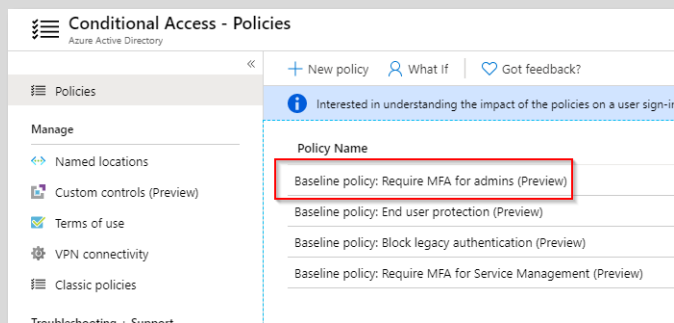
1. Select + **New Policy**
2. Go to Assignments > Users and groups > Include > **Select users and groups** > check **Directory roles**
3. At a minimum, select the following roles: Billing admin, Conditional Access admin, Exchange admin, Global admin, Helpdesk admin, Security admin, SharePoint admin, and User admin
4. Go to Cloud apps or actions > Cloud apps > Include > select **All cloud apps** (and don't exclude any apps)
5. Under Access controls > Grant > select **Grant access** > check **Require multi-factor authentication** (and nothing else)
6. Enable policy > On
7. Create

**How will this affect my users?**

When you require MFA for your privileged roles, they will need to authenticate using at least two methods when accessing the privileged account.

**View settings** Resolved through third-party Save Ignore

- 3) You'll be taken to the Conditional Access Policies. Open the Baseline policy for Require MFA for Admins.



**Conditional Access - Policies**

Azure Active Directory

+ New policy What if Got feedback?

Interested in understanding the impact of the policies on a user sign-in?

Policy Name
<b>Baseline policy: Require MFA for admins (Preview)</b>
Baseline policy: End user protection (Preview)
Baseline policy: Block legacy authentication (Preview)
Baseline policy: Require MFA for Service Management (Preview)

Troubleshooting + Support

*In this exercise you will be implementing Baseline Policies, which have been replaced by Security Defaults. New Azure Active Directory tenants won't show the Baseline Policies. You can skip the configuration.*

- 4) Review the description and *enable* the policy. **Save** the settings.

### Baseline policy: Require M...

Policies

Name

Baseline policy: Require MFA for admins (...)

This policy requires **multi-factor authentication (MFA)** for the following directory roles:

- Global Administrator
- SharePoint Administrator
- Exchange Administrator
- Conditional Access Administrator
- Security Administrator
- Helpdesk Administrator/Password Administrator
- Billing Administrator
- User Administrator

This policy also blocks legacy authentication.

[Learn more](#)

**Enable policy**

☒ Use policy immediately

☐ Do not use policy

- 5) Repeat steps **14** and **15** for the other baseline policies.

+ New policy   What if   Got feedback?	
Interested in understanding the impact of the policies on a user sign-in? Check out the "What If" tool. →	
Policy Name	State
Baseline policy: Require MFA for admins (Preview)	On
Baseline policy: End user protection (Preview)	On
Baseline policy: Block legacy authentication (Preview)	On
Baseline policy: Require MFA for Service Management (Preview)	On

- 6) From the Search Bar, navigate to the Azure AD Security panel.

Services All 44 results

- Azure AD B2C
- Azure AD Security**
- Azure AD Authentication methods
- Azure AD Connect Health
- Azure AD Domain Services
- Azure AD Identity Protection
- Azure AD Named locations
- Azure AD Password protection
- Azure AD Risk detections
- Azure AD Risky sign-ins

Resources

No results were found.

Marketplace All 20 results

- Azure AD Identity Protection
- Azure AD Domain Services
- Azure AD Privileged Identity Management
- Azure AD Connect Health

Documentation All 1000+ results

[What is Azure Active Directory? - Azure Active Directory ...](#)

[Azure Active Directory documentation | Microsoft Docs](#)

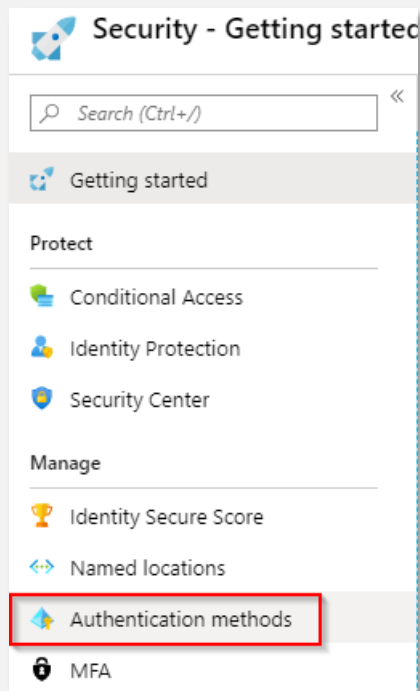
[Overview of Azure Active Directory Domain Services ...](#)

[Configure Azure Active Directory authentication - Azure ...](#)

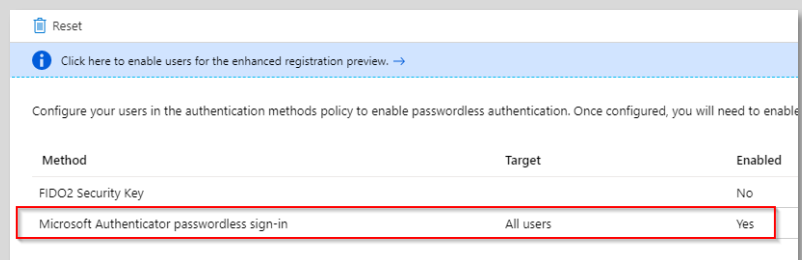
Resource Groups

No results were found.

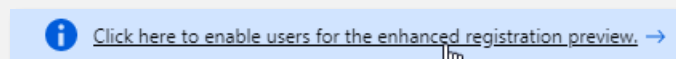
7) Browse to *Authentication Methods*.



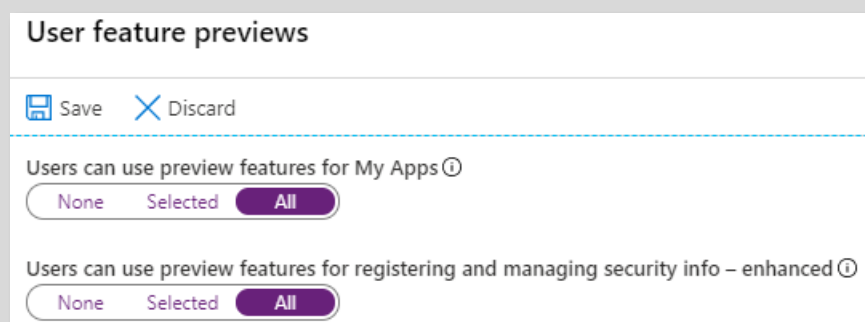
8) Enable *Microsoft Authenticator passwordless sign-in* for *All Users*.



9) Click the preview registration notification bar.



10) *Enable* all preview features and click **Save**.



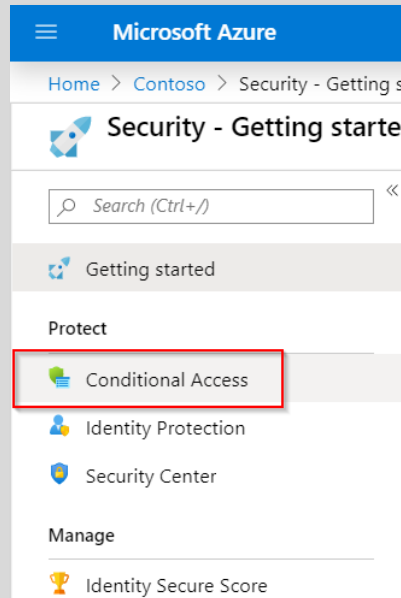
11) Make sure that you update your Authenticator



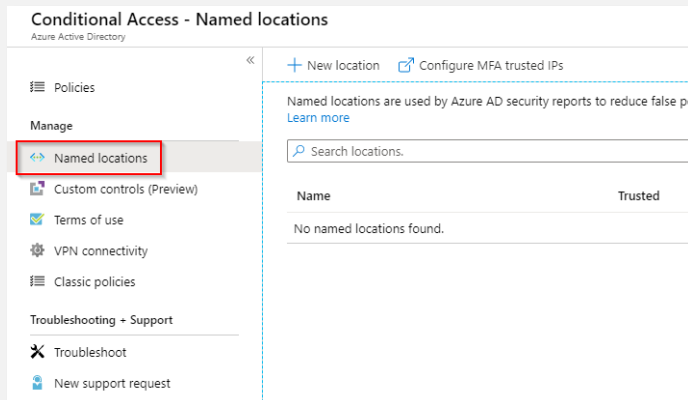
application on  
your phone.  
12) Your done for  
this exercise!

## Exercise 1d: Implement custom Conditional Access policy

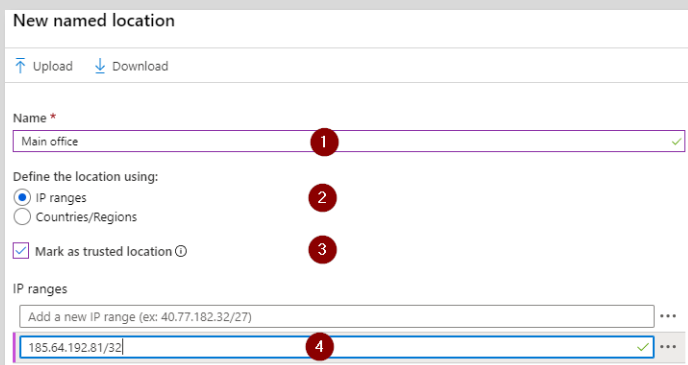
- 1) From the Azure Portal, navigate to Azure Active Directory > Security > Conditional Access



- 2) From the navigation panel, go to Named Locations.



- 3) Choose to add a New location and fill in the form. Use the IP-address you retrieved earlier.



- 4) Create a new policy and define the settings following the instructions.

New

Info

Name \*

Only allow Azure Portal from HQ 1 ✓

Assignments

Users and groups ① 2 >

All users

Cloud apps or actions ① 3 >

1 app included

Conditions ① 4 >

1 condition selected

Access controls

Grant ① 5 >

Block access

Session ① >

0 controls selected

Enable policy 6

Report-only On Off

- 5) Select the Azure Portal as a condition for this policy.

Cloud apps or actions

Select what this policy applies to

Cloud apps User actions

Include Exclude

None

All cloud apps

Select apps

Select

Microsoft Azure Management >

MA Microsoft Azure Manage... ...

6) We want every location to be included...

### Conditions

**Info**

Sign-in risk ⓘ  
Not configured

Device platforms ⓘ  
Not configured

**Locations ⓘ**  
Any location and all trusted locat...

Client apps (Preview) ⓘ  
Not configured

Device state (Preview) ⓘ  
Not configured

### Locations

Control user access based on their physical location. [Learn more](#)

**Configure ⓘ**

**Yes** No

**Include** Exclude

☒ Any location  
☐ All trusted locations  
☐ Selected locations

Select  
None

7) Except the trusted location we

### Locations

Control user access based on their physical location. [Learn more](#)

**Configure ⓘ**

**Yes** No

**Include** **Exclude**

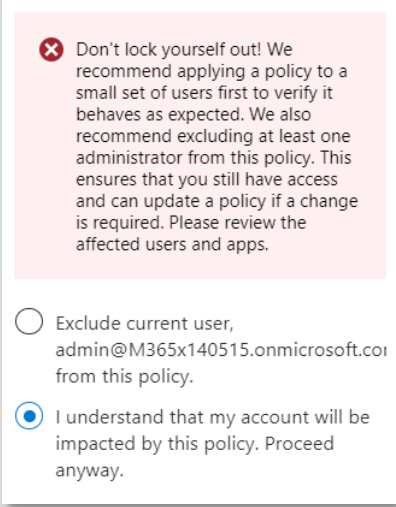

Select the locations to exempt from the policy

☒ All trusted locations  
☐ Selected locations

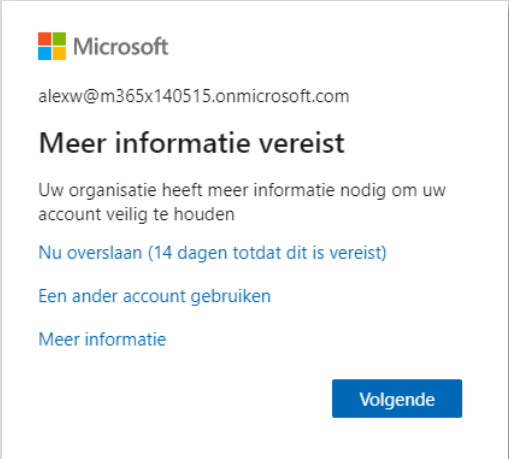
Select  
None

8) Apply the changes by clicking Done twice.

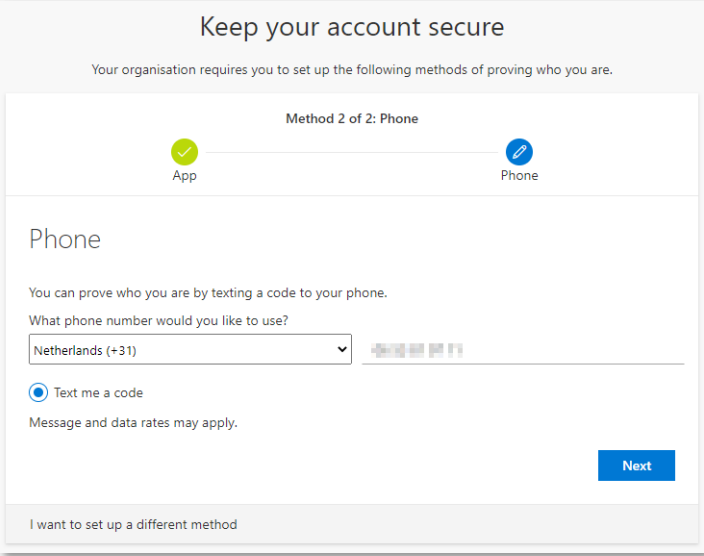
**Done**

<p>9) The Azure Portal warns on locking yourself out. As we choose to report-only, we can safely ignore the suggested exclusion.</p>	 <p>A warning dialog box with a red 'x' icon. The text reads: "Don't lock yourself out! We recommend applying a policy to a small set of users first to verify it behaves as expected. We also recommend excluding at least one administrator from this policy. This ensures that you still have access and can update a policy if a change is required. Please review the affected users and apps." Below this are two radio button options: "Exclude current user, admin@M365x140515.onmicrosoft.com from this policy." (unselected) and "I understand that my account will be impacted by this policy. Proceed anyway." (selected).</p>
<p>10) Create the policy.</p>	 <p>A blue button with the text "Create" in white.</p>

## Exercise 1e: Validate Passwordless Sign-in

<p>11) Switch back to the in-private browser window sign in again by using the <a href="#">user account</a>.</p>	 <p>A Microsoft sign-in page. At the top is the Microsoft logo and the email address "alexw@m365x140515.onmicrosoft.com". The heading is "Meer informatie vereist". Below it, the text says "Uw organisatie heeft meer informatie nodig om uw account veilig te houden". There are three links: "Nu overslaan (14 dagen totdat dit is vereist)", "Een ander account gebruiken", and "Meer informatie". At the bottom right is a blue button labeled "Volgende".</p>
--	---

- 12) Please note you're required to register for MFA. Follow the instructions to download and activate the Microsoft Authenticator app. You'll be asked to add a phone number as a secondary method. Please activate your mobile phone number.



Keep your account secure

Your organisation requires you to set up the following methods of proving who you are.

Method 2 of 2: Phone

App Phone

Phone

You can prove who you are by texting a code to your phone.

What phone number would you like to use?

Netherlands (+31)

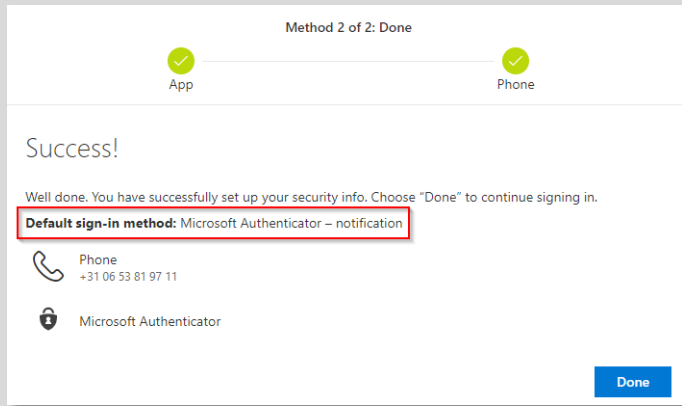
☒ Text me a code

Message and data rates may apply.

Next

I want to set up a different method

- 13) When done, both the Authenticator should be the default sign-in method.



Method 2 of 2: Done

App Phone

Success!

Well done. You have successfully set up your security info. Choose "Done" to continue signing in.

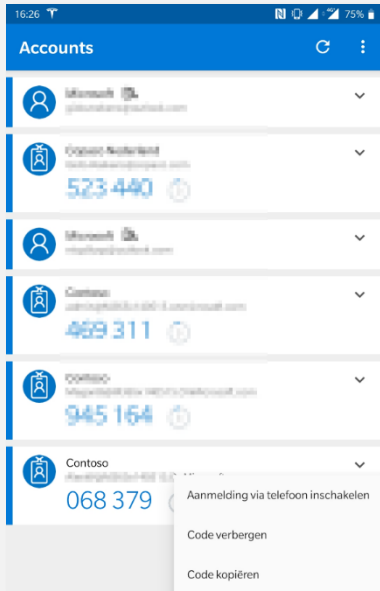
**Default sign-in method: Microsoft Authenticator - notification**

Phone  
+31 06 53 81 97 11

Microsoft Authenticator

Done

- 14) From the Authenticator app on your phone, start the Phone Sign-In registration.



Accounts

Microsoft  
g.j.van@copaco.nl

Contoso Nederland  
523 440

Microsoft  
m.van@copaco.nl

Contoso  
469 311

Contoso  
945 164

Contoso  
068 379

Aanmelding via telefoon inschakelen

Code verbergen

Code kopiëren

- 15) Sign in using the user credentials and register the

device for  
Phone Sign-In.

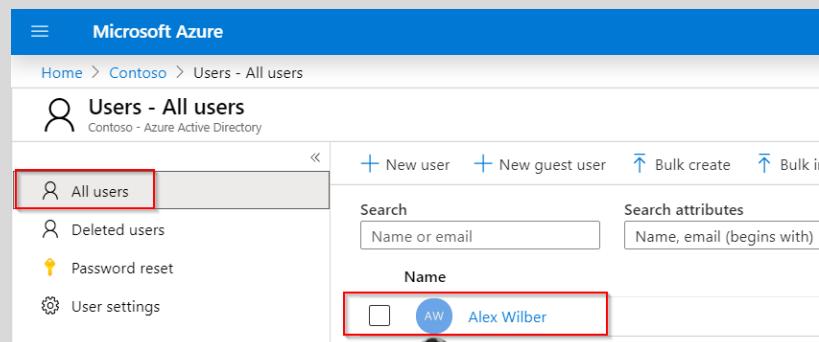
- 16) From the web browser, sign out from the account. Sign right back in using the same user account.



- 17) Please note you won't be prompted for a password. Instead, the Authenticator app now validates using a numeric value. Choose the corresponding number to validate.

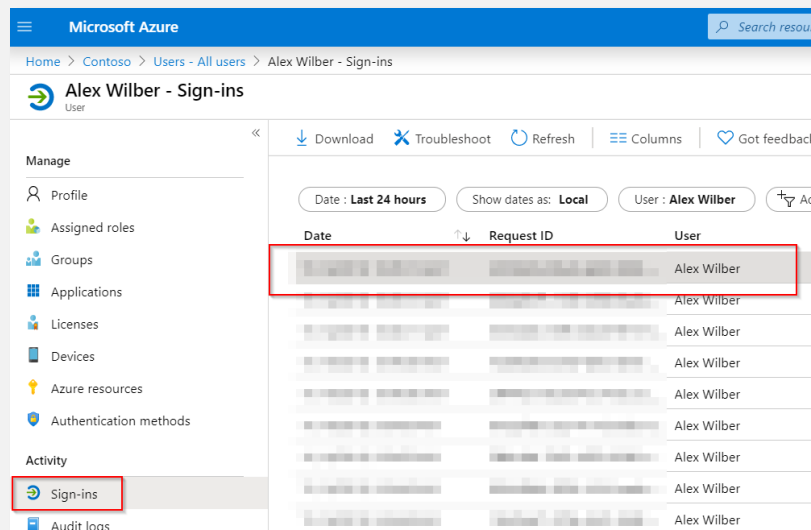
- 18) Switch back to the browser instance with the Azure Portal

- 19) Navigate to Azure Active Directory > Users and select Alex Wilber

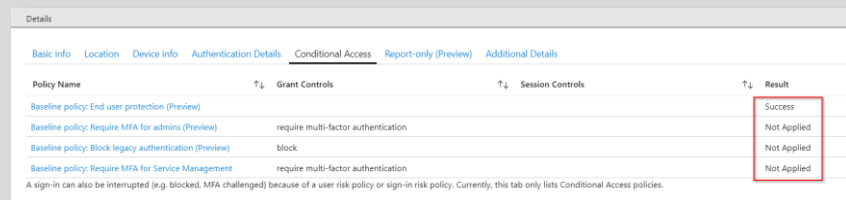




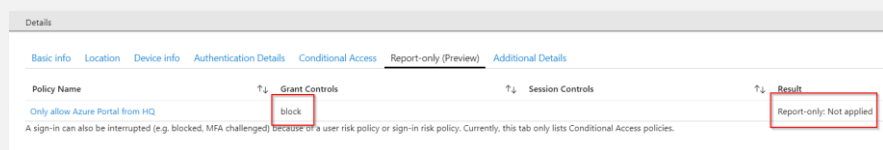
20) Select the sign-in that corresponds with your last sign-in as Alex. Wait and refresh if it's not being displayed yet.



21) Browse to the Conditional Access tab. Validate if the policies are applied as expected. This helps you understand behavior and troubleshooting concerning sign-ins.



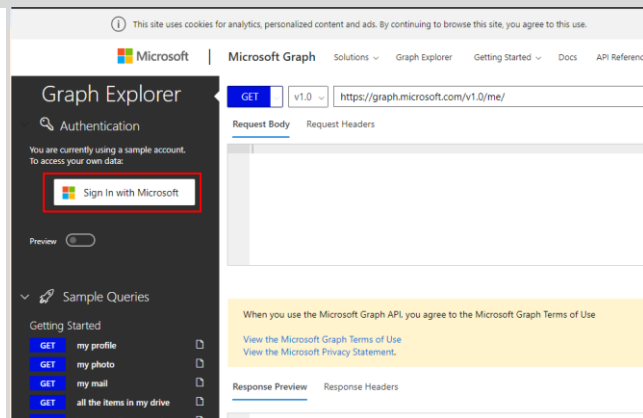
22) Browse to the Report-Only tab. Verify that the policy should block access, but hasn't because of the report-only setting. This is a great way to check your policies before taking them into production.



23) Your done for this exercise!

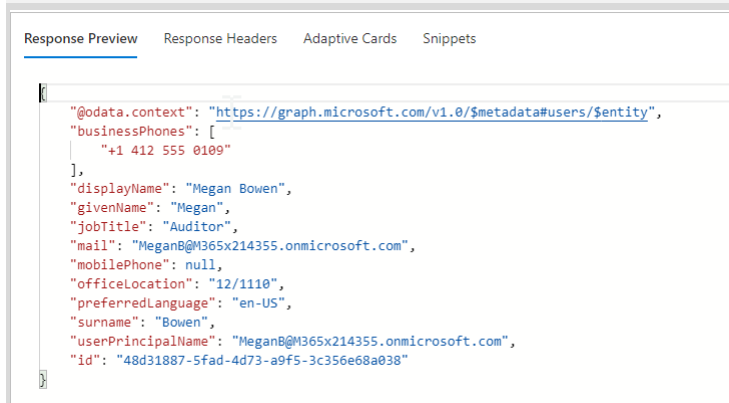
## Activity 2 : Graph API Explorer

- 1) Go to <https://developer.microsoft.com/en-us/graph/graph-explorer>
- 2) Sign in with your demo administrator account
- 3) If prompted, provide consent for the access



- 4) There is a preview version available, you can enable it to view the latest features.

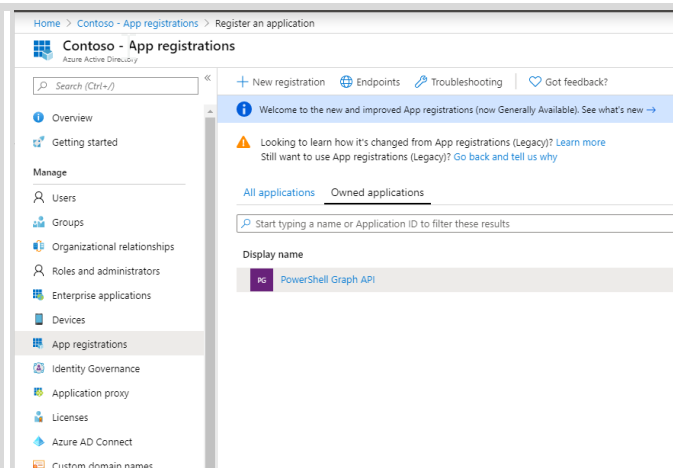
- 5) At the top of the screen, choose Run Query.
- 6) Review the output



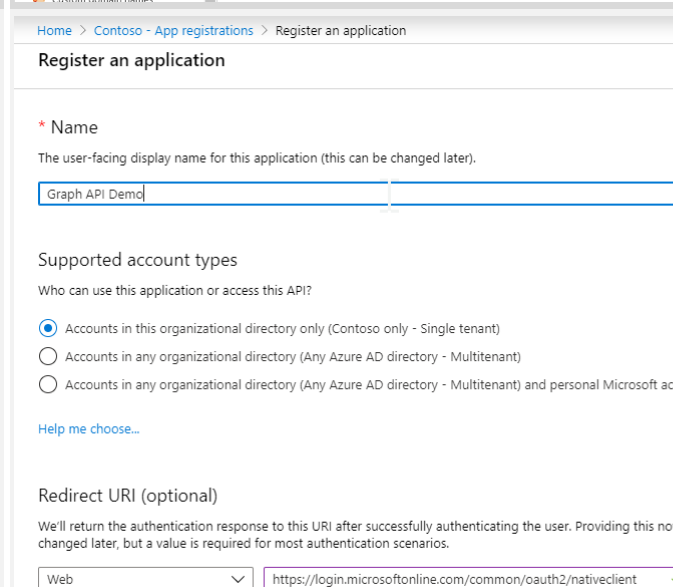
- 7) Also make note of the Permissions tab. This is an easy way to discover what permissions are required for the query.

## Activity 3 : Connect with the Graph API

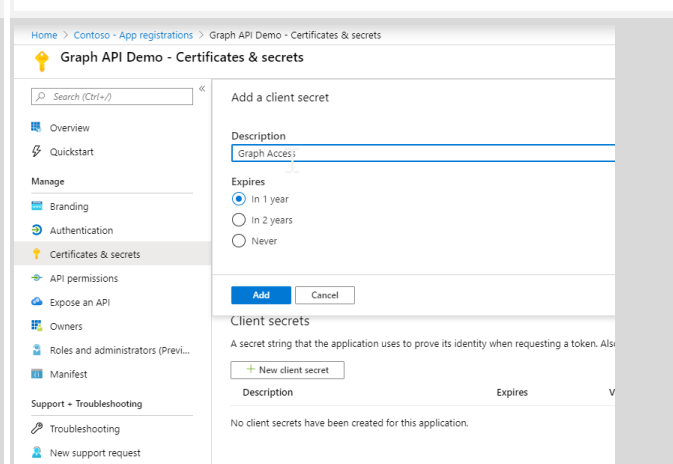
- 1) Log in to portal.azure.com
- 2) Go to Azure Active Directory
- 3) Choose App Registrations
- 4) Choose New Registration



- 5) Give the Application a Name
- 6) Choose Single Tenant
- 7) For the Redirect URI enter <https://login.microsoftonline.com/common/oauth2/nativeclient>
- 8) Finally click Register

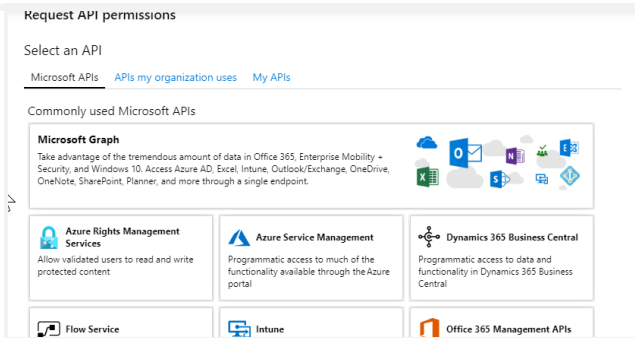


- 9) Go to Certificates & secrets and choose New Client Secret
- 10) Choose 1 Year and give it an Description
- 11) Choose Add
- 12) **Copy the Client Secret value** you wont be able to get it after you leave the page. And store it some where save

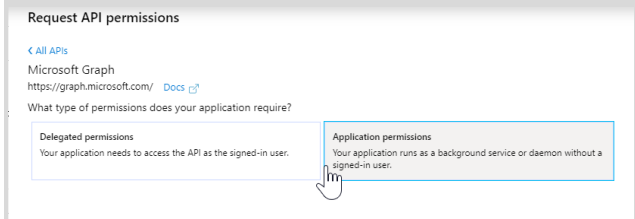




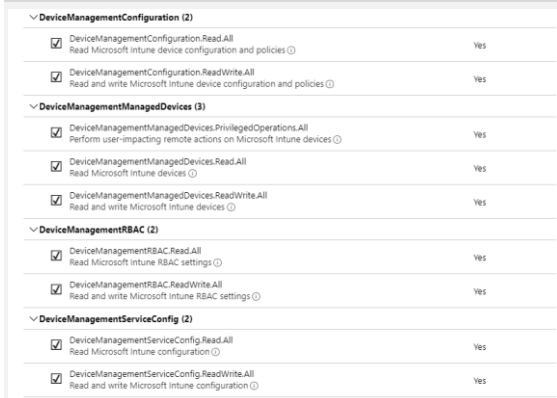
- 13) Go to API permissions
- 14) Choose Add a permission
- 15) Select Microsoft Graph



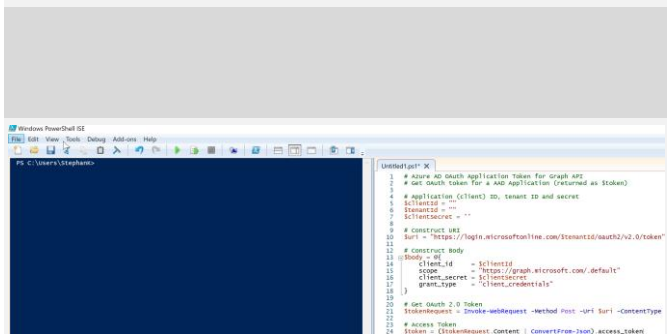
- 16) Choose Application permissions (for the demo this is easiest)



- 17) You can now select which permissions are needed.
- 18) Select all user and devices permissions and choose add permissions
- 19) Choose Add permissions



- 20) Scroll to the bottom of the screen and select Grant admin consent for "name of your tenant" and choose yes



- 21) Open PowerShell ISE, and copy the script Get-AuthToken.ps1 in the Script Pane

- 22) Go to [https://portal.azure.com/#blade/Microsoft\\_AAD\\_IAM/ActiveDirectoryMenuBlade/RegisteredApps](https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/RegisteredApps) and select the app that you have registered. In the overview you can find the Application ID, Directory ID





## Activity 4 : Custom Graph API commands

1) From GitHub copy the Lab Graph cmds in a new page in PowerShell ISE	
2) Make sure to adjust the parameters that match your tenant <b>+ some of the scripts assume that you have configuration policies and compliance policies in place. If you don't have these in the tenant create some!</b>	
3) Get all users	<pre>#get all users \$uri = "https://graph.microsoft.com/beta/users" \$users = Invoke-RestMethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer " \$users.value \$users.value   Select-Object DisplayName, ID, UserPrincipalName</pre>
4) Get specific user 5) Make sure that you adjust the upn for the specific	<pre>#get specific user \$uri = "https://graph.microsoft.com/beta/users/CameronW@M365x428595.onmicrosoft.com" \$megan = invoke-restmethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer " \$megan.displayName \$megan   Select-Object DisplayName, MobilePhone, City</pre>
6) Update user info	<pre>#update user info \$PatchJSON = @{"mobilephone" = "+31640409642" "city" = "Eindhoven" }   ConvertTo-Json Invoke-RestMethod -Uri \$uri -Method PATCH -Headers @{Authorization = "Bearer " -Body \$PatchJSON</pre>
7) Check updated info 8) Make sure that you adjust the upn for the specific	<pre>#Check if user info is updated \$uri = "https://graph.microsoft.com/beta/users/CameronW@M365x428595.onmicrosoft.com" \$megan = invoke-restmethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer " \$megan.displayName \$megan   Select-Object DisplayName, MobilePhone, City</pre>
9) Create new user 10) Make sure that you adjust the upn your tenant	<pre>#create new user \$uri = "https://graph.microsoft.com/beta/users" \$NewUserJSON = @{"accountEnabled" = \$true "displayName" = "EL Demo User" "mailNickname" = "eldemouser" "userPrincipalName" = "eldemouser@M365x428595.onmicrosoft.com" "mobilephone" = "+31640409642" "city" = "Eindhoven" "passwordProfile" = @{"forceChangePasswordNextSignIn" = \$true</pre>
11) Check created user	<pre>#check created user \$uri = "https://graph.microsoft.com/beta/users" \$DemoUser = invoke-restmethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer " \$DemoUser.displayName \$DemoUser   Select-Object DisplayName, MobilePhone, City</pre>
12) Delete created user	<pre>#get and delete created user \$uri = "https://graph.microsoft.com/beta/users" \$uri = \$uri + '/' + \$response.id invoke-restmethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer " Invoke-RestMethod -Method DELETE -Uri \$uri -Headers @{Authorization = "Bearer "</pre>



13) Get all groups	<pre>#get all groups \$uri = "https://graph.microsoft.com/beta/groups" Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers \$groups = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers \$groups.value   ft DisplayName</pre>
14) Get Member of first group	<pre>#get member of first group \$groups.value[0] \$groupid = \$groups.value[0].id \$uri = \$uri + '/' + \$groupid Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers \$uri = \$uri + '/' + 'members' \$members = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers</pre>
15) Get groups a user is member of	<pre>#get groups user is member of \$uri = "https://graph.microsoft.com/beta/users/\$userid/groups" \$membership = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers \$membership.value   select DisplayName</pre>
16) Export Device configuration Profiles	<pre>1 2 ###Export Device configuration profiles 3 \$uri = "https://graph.microsoft.com/beta/deviceManagement/policies" 4 \$configs = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers 5 6 foreach (\$config in \$configs.value) { 7     \$configname = \$config.displayName 8     \$configfile = "C:\temp\\$configname" + '.json' 9     \$config   ConvertTo-Json   out-file \$configfile 10 } 11</pre>
17) Export Device compliance policies	<pre>###Export Device Compliance Policies \$uri = "https://graph.microsoft.com/beta/deviceManagement/policies/compliancePolicies" \$compliances = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers  foreach (\$compliance in \$compliances.value) {     \$configname = \$compliance.displayName     \$configfile = "C:\temp\\$configname" + '.json'     \$config   ConvertTo-Json   out-file \$configfile }</pre>
18) Hello For Business settings	<pre>###Export Hello For Business Settings \$uri = "https://graph.microsoft.com/beta/deviceManagement/HelloForBusiness" \$WHBusiness = Invoke-RestMethod -Method GET -Uri \$uri -Header \$headers  \$configname = \$WHBusiness.value   select Displayname \$configfile = "C:\temp\\$(\$configname[0])" + '.json' \$config   ConvertTo-Json   out-file \$configfile</pre>
19) Import Device Configuration Policies 20) Download the new-policy-demo.json from GitHub to your local pc. 21) Edit the \$newPolicy variable so that the json is imported to the variable 22) Run the script. 23)	<pre>#Import Device Configuration Policy \$uri = "https://graph.microsoft.com/beta/deviceManagement/deviceconfigurations" \$newPolicy = get-content "C:\Users\Stephank\Desktop\MSIX demo\Graph Api\new-policy-demo.js"  \$OutputNewPolicy = Invoke-RestMethod -Method POST -Uri \$uri -Headers @{Authorization = "Bearer \$token"} \$uri = "https://graph.microsoft.com/beta/deviceManagement/deviceconfigurations/\$(\$OutputNewPolicy.id)" Invoke-RestMethod -Method GET -Uri \$uri -Headers @{Authorization = "Bearer \$token"} -ErrorAction SilentlyContinue</pre>

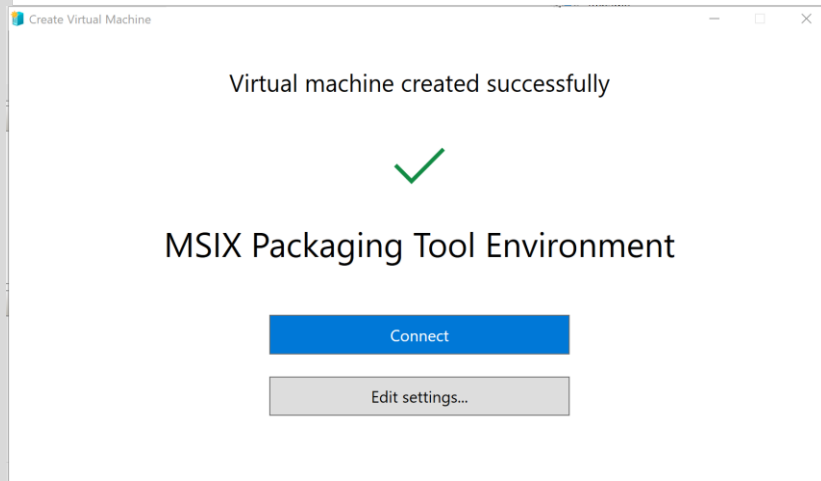
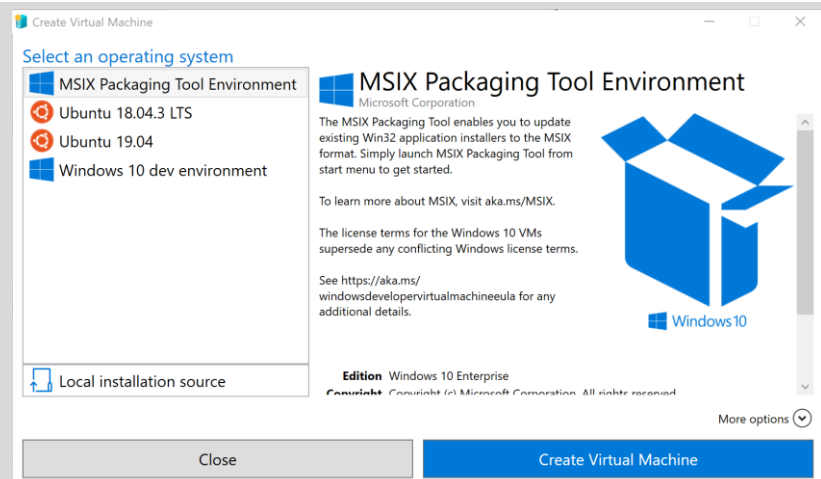


- 24) Check the Intune Portal to confirm that a new Configuration Policy is created.

## Activity 5 : MSIX

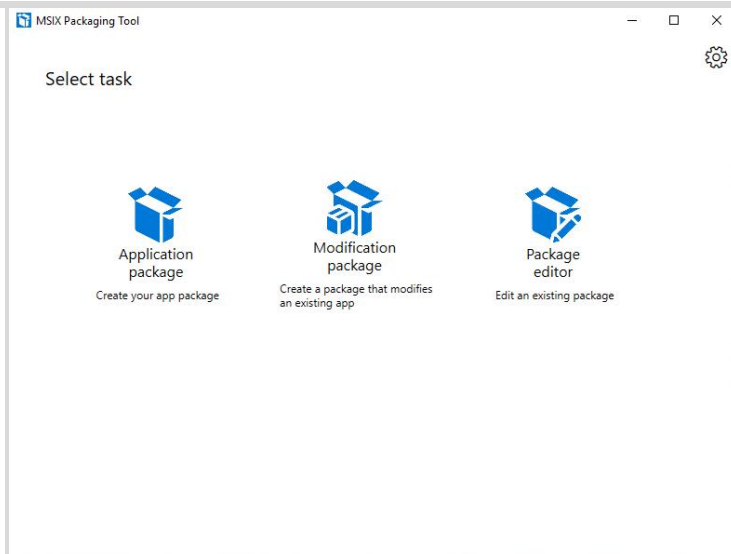
### Exercise 1a: Create Packaging VM

- 1) Open Hyper-V manager
- 2) Right click your host name and choose Quick Create
- 3) Pick MSIX Packaging Tool Environment and choose **Create Virtual Machine**. This will start to download the VM for you
- 4) When the VM is ready you can connect to the VM

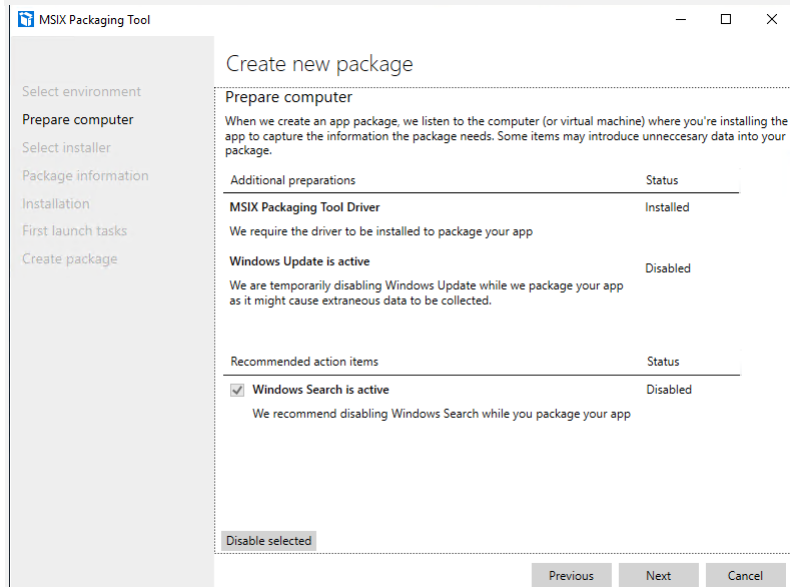


- 5) Start the VM and go through the installation process. Choose to domain join instead of a Microsoft account
- 6) When the VM is ready to use, it best that you create a snapshot (check point) so you can revert to this clean state in the future.
- 7) Download the Citrix receiver app and the pfx certificate from GitHub
- 8) Copy the installation files for the Citrix Receiver and the certificate to the VM. The files are available on GitHub

- 9) Start the MSIX Packaging tool
- 10) Choose Application Package

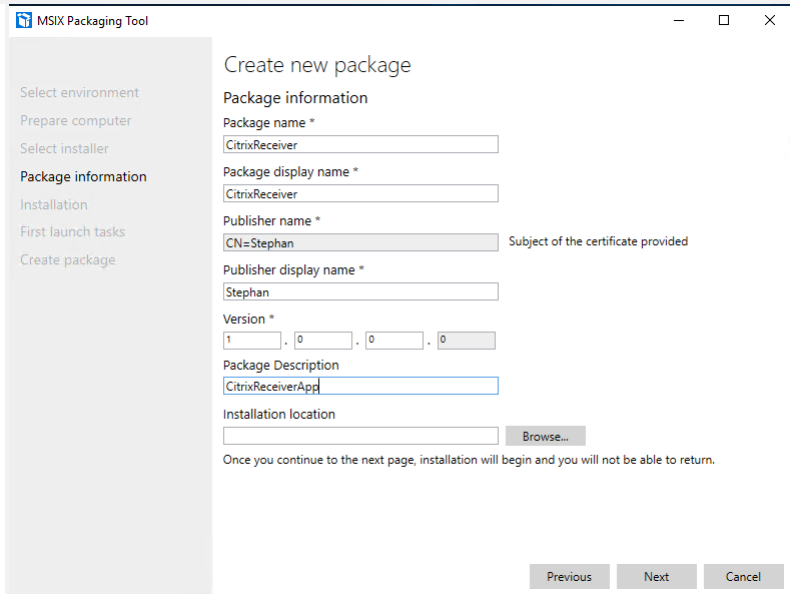


- 11) Create Package on this computer and click next.
- 12) Disable Windows Search and click next.



- 13) Browse to the Citrix receiver installation file
  - 14) At Signing preference choose **Sign with a certificate .pfx**. Browse to the certificate. The password is: **Welkom123\$**
- Choose Next

- 15) Enter the information for the application
- 16) Because the certificate uses the name **Stephan**, you must use this for the publisher display name. Choose next

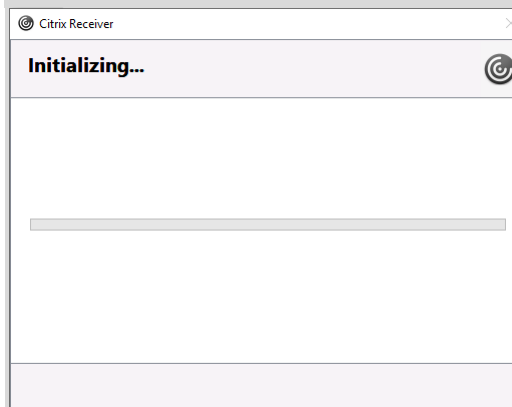


The screenshot shows the 'MSIX Packaging Tool' window with the 'Create new package' wizard. The 'Package information' tab is active. The fields are filled as follows:

- Package name \*: CitrixReceiver
- Package display name \*: CitrixReceiver
- Publisher name \*: CN=Stephan (Subject of the certificate provided)
- Publisher display name \*: Stephan
- Version \*: 1.0.0.0
- Package Description: CitrixReceiverApp
- Installation location: (empty) [Browse...]

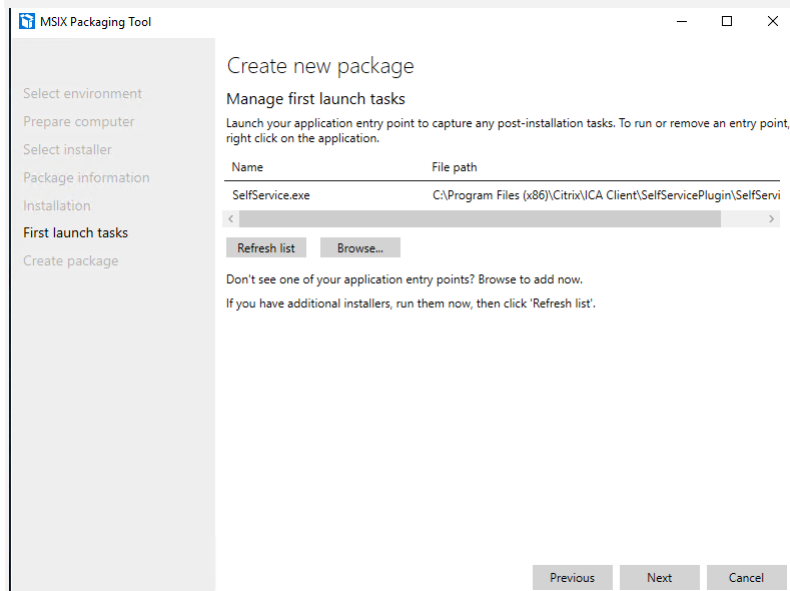
At the bottom, there are 'Previous', 'Next', and 'Cancel' buttons. A warning message at the bottom states: 'Once you continue to the next page, installation will begin and you will not be able to return.'

- 17) The Citrix receiver installation start now.
- 18) Following the installation steps
- 19) When the installation is done choose Finish
- 20) In the MSIX console choose Next



The screenshot shows the 'Citrix Receiver' window with the title 'Initializing...'. It features a progress bar and a circular loading icon. The window is currently empty except for the header and progress bar.

- 21) For now we can skip this, but for advanced installations you can continue to configure the first launch steps.
- 22) Choose next



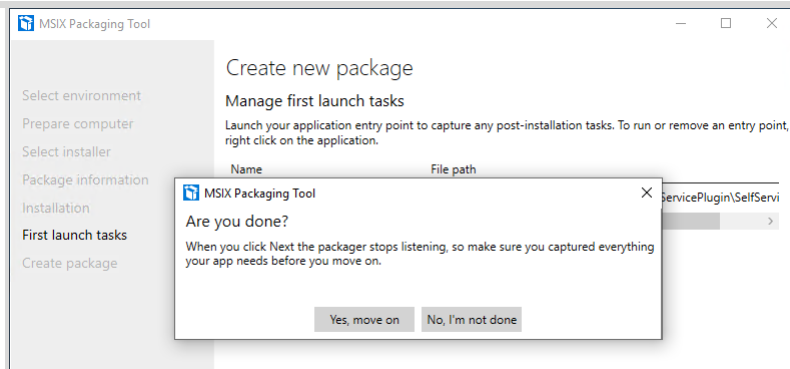
The screenshot shows the 'MSIX Packaging Tool' window with the 'Create new package' wizard. The 'Manage first launch tasks' tab is active. It displays a table with application entry points:

Name	File path
SelfService.exe	C:\Program Files (x86)\Citrix\ICA Client\SelfServicePlugin\SelfServi

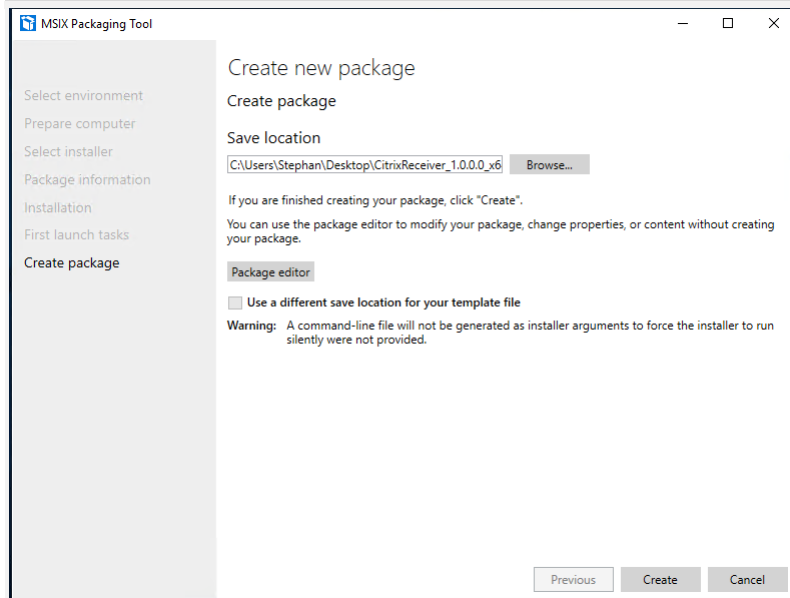
Below the table are 'Refresh list' and 'Browse...' buttons. A message at the bottom says: 'Don't see one of your application entry points? Browse to add now. If you have additional installers, run them now, then click 'Refresh list'.'

At the bottom, there are 'Previous', 'Next', and 'Cancel' buttons.

- 23) Yes, move on
- 24) Your package is created



- 25) Choose a save location, and choose **Create**



## Exercise 1b: Test your package

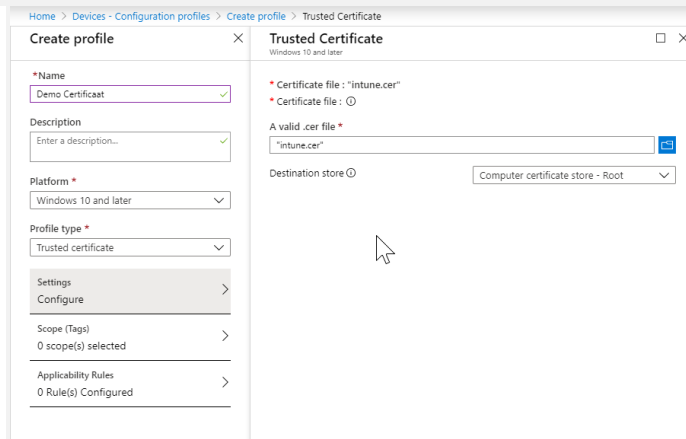
Now you created the package you probably want to test it. There are a couple of ways you can do this. The first (and most easy way to do this) is to use the same MSIX machine you created. Since this being an M365 lab we can also deploy the MSIX with Intune, but this requires you to have a test VM which you can manage with Intune. This lab won't describe how to set up the test VM but you are free to set it up.

### Deploy the MSIX application with Intune

Because the MSIX is signed with an self signed certificate that isn't trusted by default, we first have to deploy the certificate

- 1) Go to <https://devicemanagement.microsoft.com/> and to Devices. Here you can create a new Configuration Profile

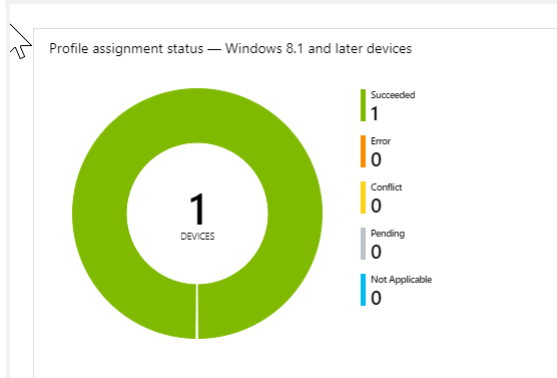
- 2) Give the configuration a Name
- 3) For Platform choose **Windows 10 and later**
- 4) For Profile Type pick **Trusted certificate**
- 5) When you select Configure you can choose the cer file. This cer file can also be found at GitHub
- 6) For the Destination store you must choose **Computer certificate store – Root**
- 7) Choose OK and Create



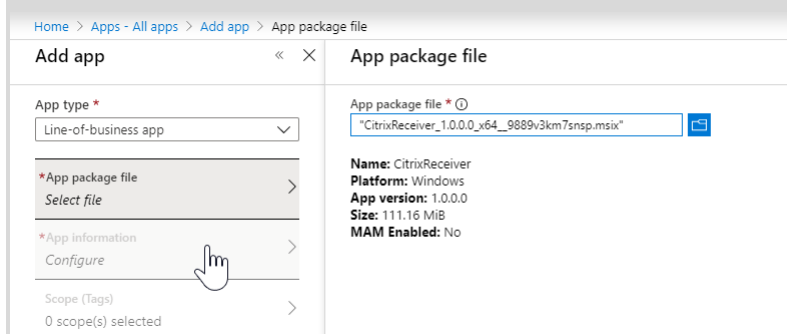
The screenshot shows two side-by-side windows. The 'Create profile' window on the left has the following settings: Name 'Demo Certificaat', Description 'Enter a description...', Platform 'Windows 10 and later', Profile type 'Trusted certificate', and 'Settings Configure' expanded. The 'Trusted Certificate' window on the right shows: Certificate file 'intune.cer', A valid .cer file 'intune.cer', and Destination store 'Computer certificate store - Root'.

- 8) When the certificate is uploaded you can assign the Profile to a group which has the test VM in it.

- 9) **Wait and verify till the certificate is successfully is deployed to your test VM**

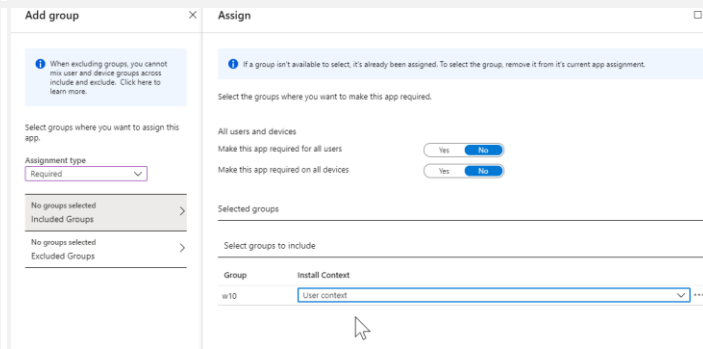


- 10) When the certificate is successfully deployed you can continue to deploy the Citrix receiver application
- 11) In the device management portal go to **Apps → All apps** and choose **Add**
- 12) App type is **Line-of-Business app**
- 13) At App package file navigate to your Citrix receiver MSIX package
- 14) Choose OK
- 15) Open the App information settings and reviews the settings. When ready choose OK
- 16) Choose Add
- 17) The application is now being uploaded. When ready you can continue to the next step



The screenshot shows two side-by-side windows. The 'Add app' window on the left has: App type 'Line-of-business app', App package file 'Select file', App information 'Configure', and Scope (Tags) '0 scope(s) selected'. The 'App package file' window on the right shows: App package file 'CitrixReceiver\_1.0.0.0\_x64\_9889v3km7snp.msix', Name 'CitrixReceiver', Platform 'Windows', App version '1.0.0.0', Size '111.16 MiB', and MAM Enabled 'No'.

- 18) Go to assignment and choose **Add Group**
- 19) For assignment type choose **Required**
- 20) Pick the group which has your test vm in it and install in the **User Context**
- 21) Click Save



22) Wait for the deployment to finish

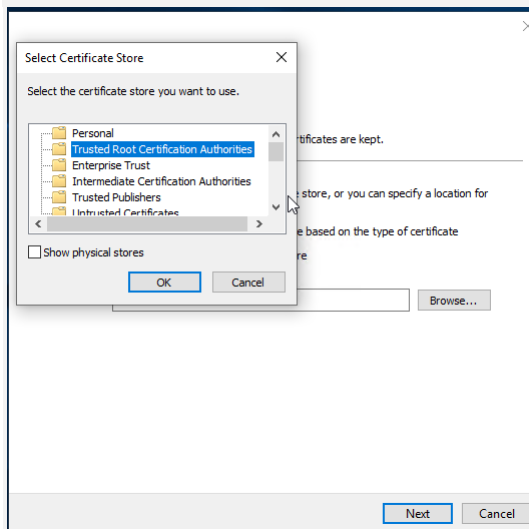
23)

### Test the Package locally

If you don't have a test VM which is enrolled in Intune you can also use the MSIX packaging machine to test your package.

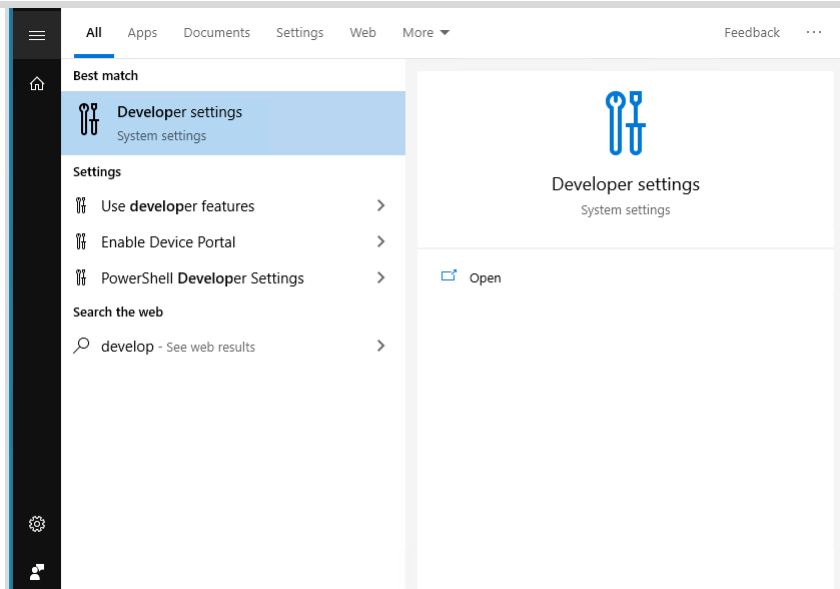
- 1) Copy the MSIX package you created to your local PC.
- 2) Revert to the clean state of the VM by applying the checkpoint you created
- 3) Copy the MSIX package and the certificate to the machine

- 4) Install the certificate by double clicking the certificate
- 5) Choose Local Machine
- 6) Next
- 7) Password is **Welkom123\$**, and choose next
- 8) Choose Place all certificates in the following store
- 9) Select **Trusted Root Certification Authorities**
- Next
- 10) Finish

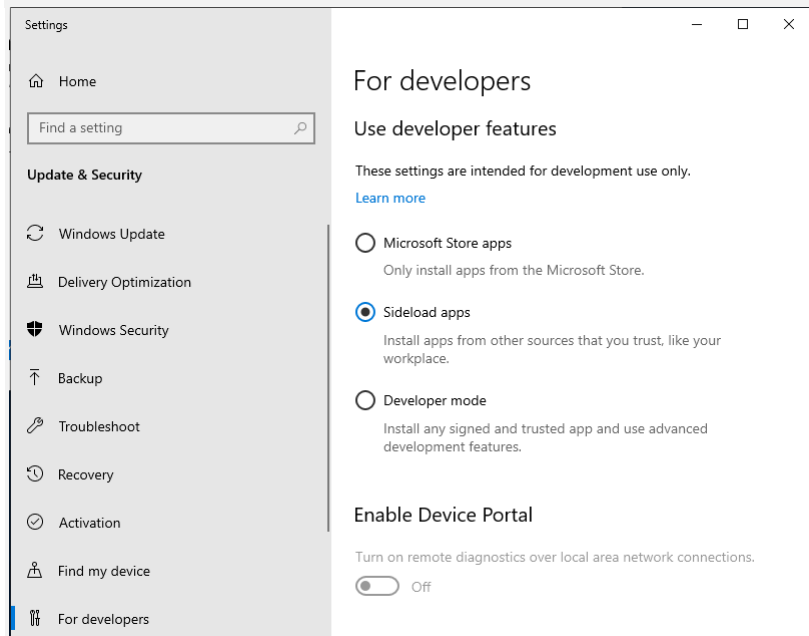




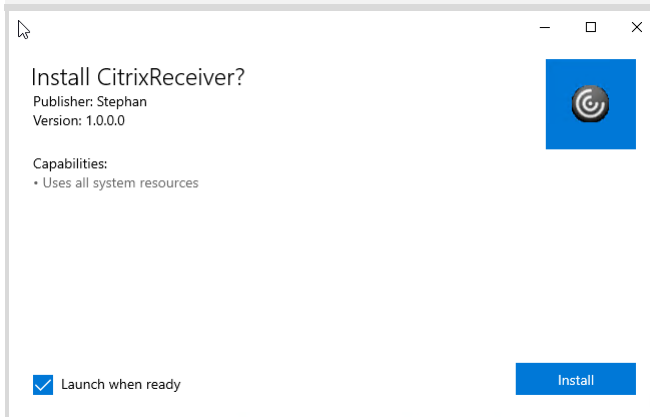
11) From the start menu search for **Developer settings**



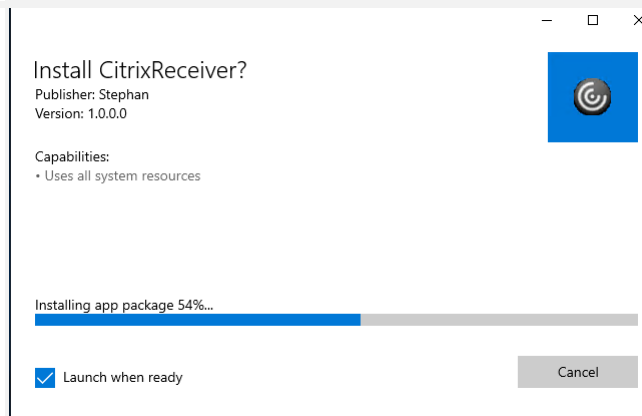
12) Choose Sideload apps



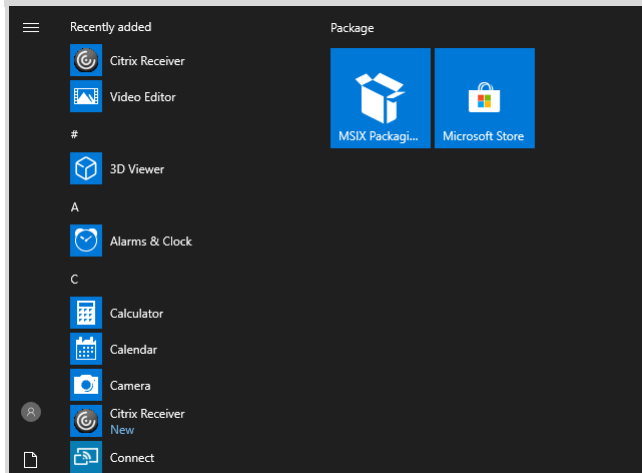
13) Then Double click your MSIX package  
14) Choose Install



15) The applications is being installed



16)



17)

## Extra resources