

# Modern Workplace Advanced

**Azure Hands On Lab February 18<sup>th</sup> 2020**

V4.0 by :      Stephan van de Kruis  
                         Gido Weekens



2tCloud



## Content

Introduction .....	3
Activity 1 : Use Passwordless Sign-in to prevent phishing.....	4
Exercise 1a : Setting up the user accounts .....	4
Exercise 1b: Implement Passwordless Sign-in .....	7
Exercise 1c: Implement custom Conditional Access policy .....	9
Exercise 1d: Validate Passwordless Sign-in.....	13
Activity 2 : Prevent against phishing attacks with Defender for Office 365.....	17
Activity 3 : Graph API.....	20
Exercise 4b : Graph API Explorer .....	20
Exercise 4b : Connect with the Graph API .....	21
Exercise 4c : Custom Graph API commands .....	24
Activity 6 : MSIX.....	27
Exercise 6a : Create Packaging VM.....	27
Exercise 6b : Test your package.....	30
Exercise 6c : Deploy the MSIX application with Intune .....	30
Exercise 6d : Test the Package locally .....	32

## Introduction

### *Estimated time to complete this lab*

120 minutes

### *Objectives*

During this lab, you will learn how to get started with Microsoft 365 to;

- Implement a passwordless authentication policy to prevent users from phishing
- Implement a Defender for Office 365 policy to prevent phishing
- Use the Graph API to configure a Microsoft 365 tenant
- Deploy a MSIX application package

### *Prerequisites*

To complete this course, you will be needing;

- Laptop/computer with Internet browser and Wi-Fi connected
- A Microsoft 365 business subscription

### *Materials*

All student materials are available for download here:

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



## Activity 1 : Use Passwordless Sign-in to prevent phishing

*Estimated time to complete this activity*

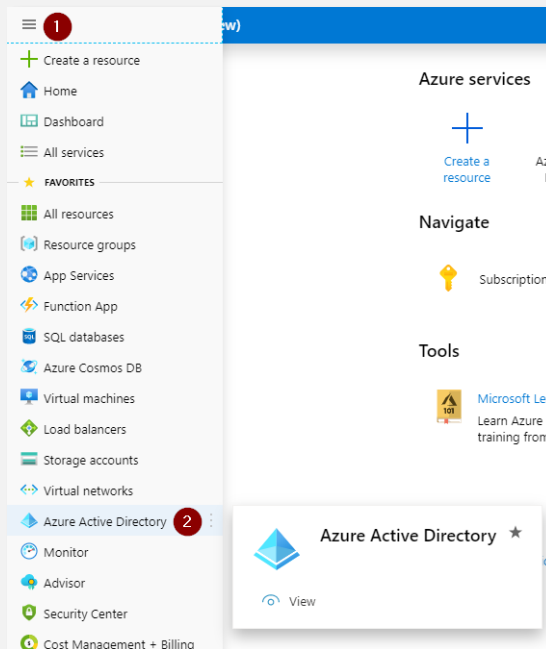
45 minutes

### 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

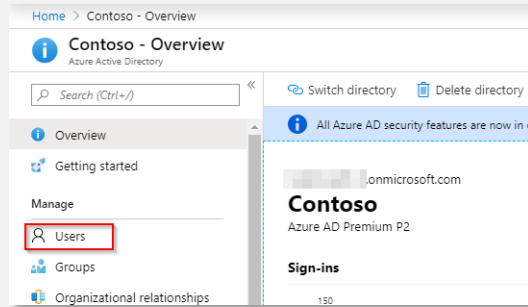
- 4) From GitHub, [download UserTemplate.csv](#) and [save](#) it to your local computer.



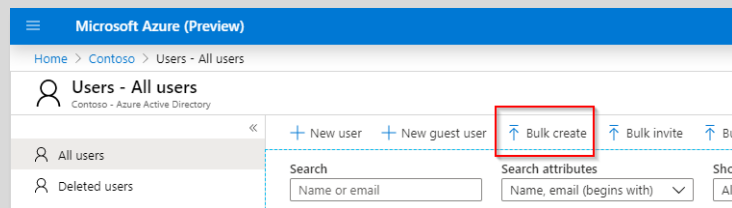
- 5) [Edit](#) the CSV-file using Notepad. Find and replace `<tenant>` so it corresponds with the tenant you're using. **Save** your changes.

```
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 Admin
Diego Siciliani,DiegoS@<tenant>.onmicrosoft.com>Password2019!,Diego,Siciliani,HR
Isaiah Langer,IsaiahL@<tenant>.onmicrosoft.com>Password2019!,Isaiah,Langer,Sales
Joni Sherman,JoniS@<tenant>.onmicrosoft.com>Password2019!,Joni,Sherman,Paralegal,
Lynne Robbins,LynneR@<tenant>.onmicrosoft.com>Password2019!,Lynne,Robbins,Planner
Megan Bowen,MeganB@<tenant>.onmicrosoft.com>Password2019!,Megan,Bowen,Marketing
Nestor Wilke,NestorW@<tenant>.onmicrosoft.com>Password2019!,Nestor,Wilke,Director
Patti Fernandez,PattiF@<tenant>.onmicrosoft.com>Password2019!,Patti,Fernandez,Pres
```

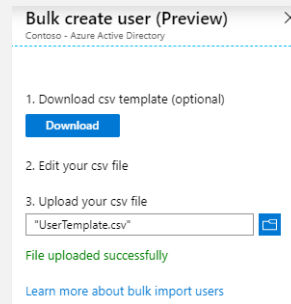
- 6) Open the [Users](#) blade



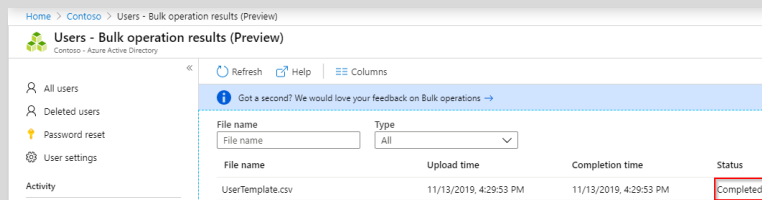
- 7) Choose [Bulk create](#)



- 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,



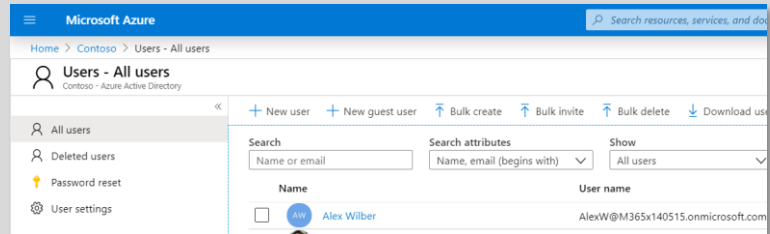
- 10) Open a new InPrivate browser window

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

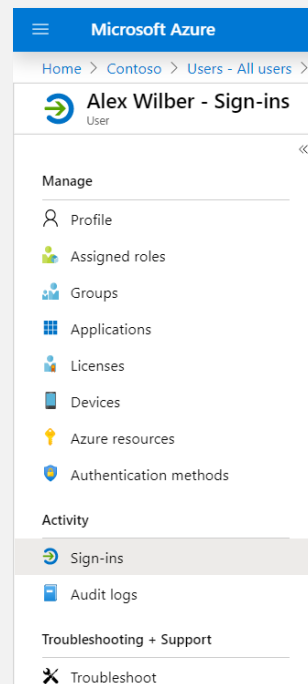
Username [AlexW@<tenant>.onmicrosoft.com](#) &  
Password [Password2019!](#)

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

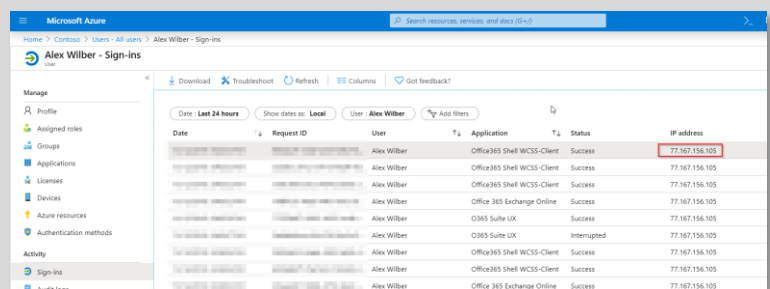
13) Browse to the Azure Active Directory and then Users



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 isn't showing yet.

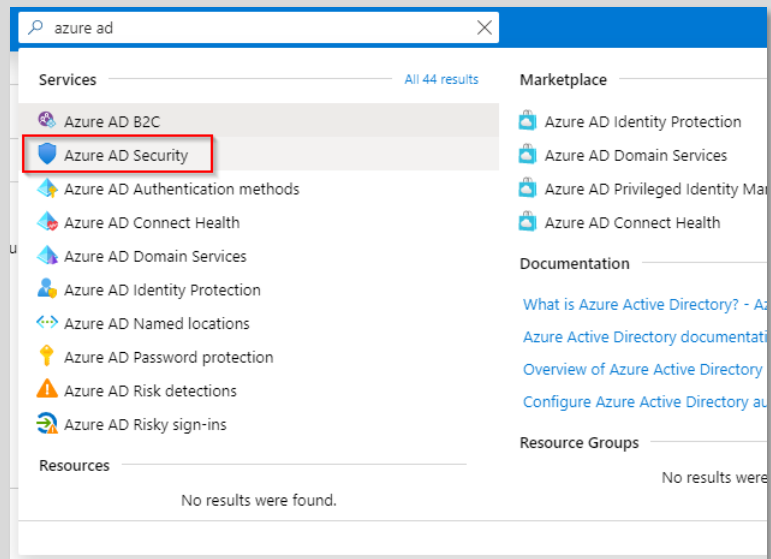


16) Take note of the IP-address that was used to perform the sign-in. You will need this later.

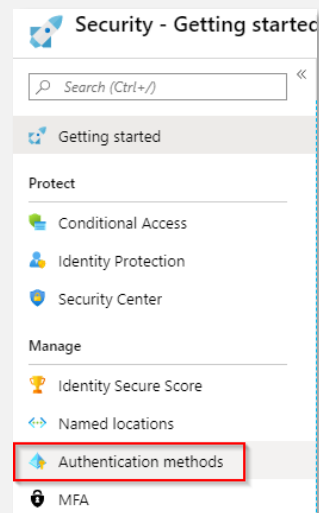
17) Your done for this exercise!

## Exercise 1b: Implement Passwordless Sign-in

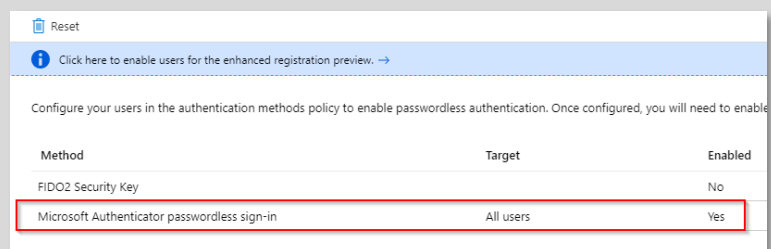
- 1) From the Search Bar, navigate to the Azure AD Security panel.



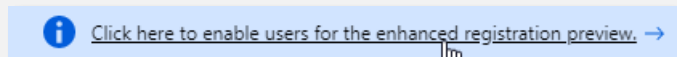
- 2) Browse to [Authentication Methods](#).



- 3) Enable [Microsoft Authenticator passwordless sign-in](#) for [All Users](#).



- 4) Click the preview registration notification bar.





- 5) **Enable** all preview features and click **Save**.

- 6) Make sure that you update your Authenticator application on your phone.

- 7) Your done for this exercise!

#### User feature previews

 Save  Discard

Users can use preview features for My Apps ⓘ

☐ None ☐ Selected ☒ All

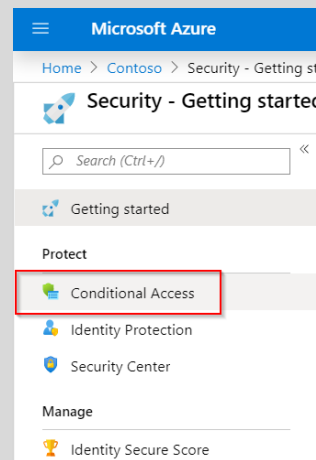
Users can use preview features for registering and managing security info – enhanced ⓘ

☐ None ☐ Selected ☒ All

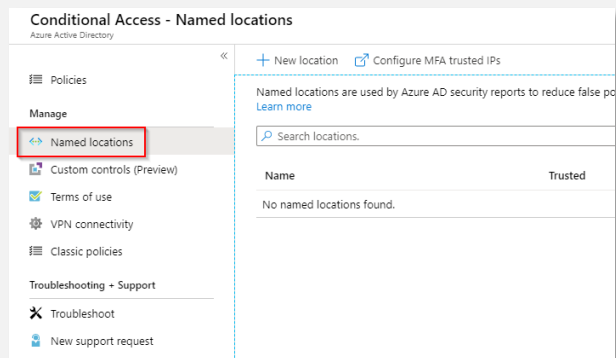


## Exercise 1c: Implement custom Conditional Access policy

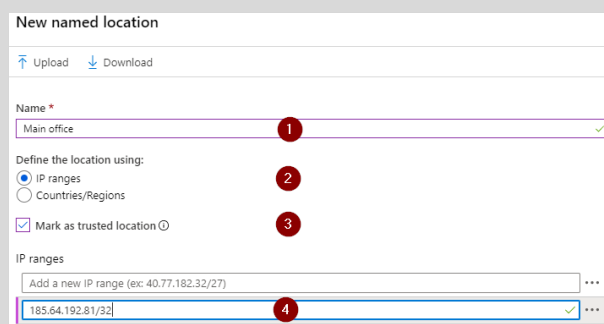
- 1) From the Azure AD Security blade, navigate to Conditional Access (or use the Search)



- 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 Conditional Access policy to require MFA for all users. Define the settings following the example on the right..

### New

Conditional access policy

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

Name \*

Require MFA for everyone 1 ✓

Assignments

Users and groups ⓘ

All users 2

Cloud apps or actions ⓘ

All cloud apps 3

Conditions ⓘ

0 conditions selected

Access controls

Grant ⓘ

1 control selected 4

Session ⓘ

0 controls selected

- 5) Make sure to Grant access with the requirement for MFA.

### Grant

Control user access enforcement to block or grant access. [Learn more](#)

☐ Block access

☒ Grant access 1

☒ Require multi-factor authentication 2

☐ Require device to be marked as compliant ⓘ

☐ Require Hybrid Azure AD joined device ⓘ

☐ Require approved client app ⓘ  
[See list of approved client apps](#)

☐ Require app protection policy ⓘ  
[See list of policy protected client apps](#)

☐ Require password change ⓘ

For multiple controls

☒ Require all the selected controls

☐ Require one of the selected controls

- 6) The Azure Portal warns on locking yourself out. We can safely ignore the suggested exclusion, as we're aware of the result.

✖ 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.

☐ Exclude current user, admin@M365x140515.onmicrosoft.com from this policy.

☒ I understand that my account will be impacted by this policy. Proceed anyway.

- 7) Create another new policy and define the settings following the instructions.

**New** □ ×

📄 Info

**Name \***

Only allow Azure Portal from HQ 1 ✓

**Assignments**

Users and groups 📄 2 >

All users

Cloud apps or actions 📄 3 >

1 app included

Conditions 📄 4 >

1 condition selected

**Access controls**

Grant 📄 5 >

Block access

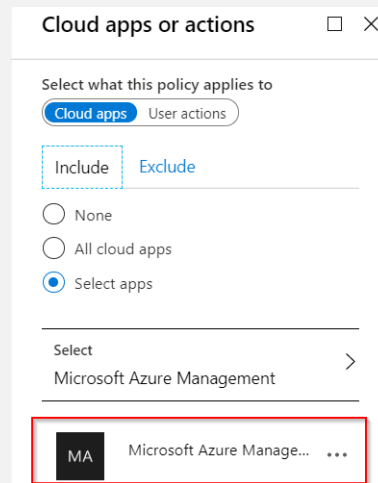
Session 📄 >

0 controls selected

**Enable policy** 6

Report-only On Off

8) 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

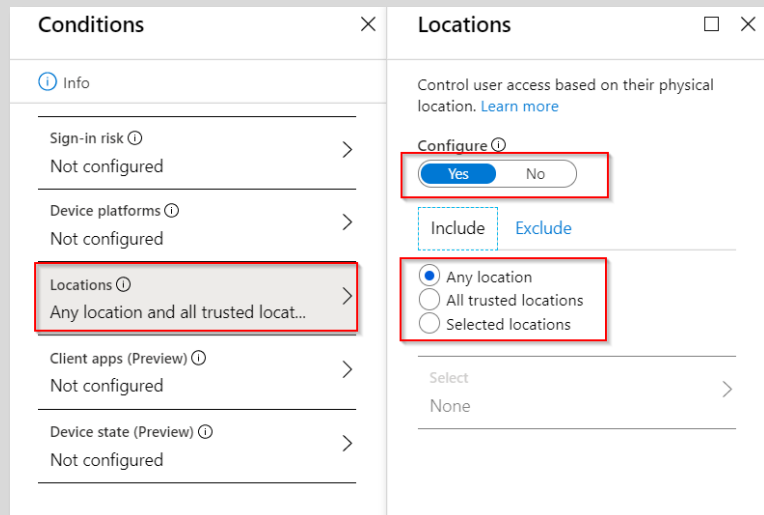
☒ Select apps

Select

Microsoft Azure Management >

**MA** Microsoft Azure Manage... ...

9) 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

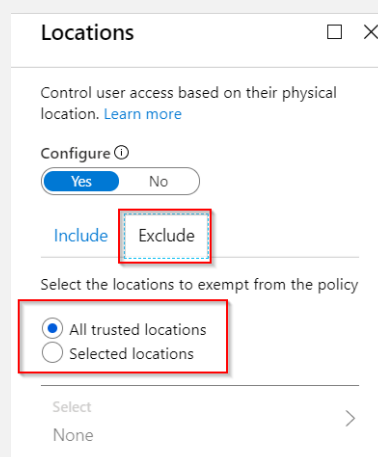
☐ All trusted locations

☐ Selected locations

Select >

None

10) Except the trusted location we defined before



**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

11) The Azure Portal warns on locking yourself out again. As we choose to report-only, we can safely ignore.

⛔ 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.

- ☐ Exclude current user, admin@M365x140515.onmicrosoft.com from this policy.
- ☒ I understand that my account will be impacted by this policy. Proceed anyway.

12) Create the policy.

Create

## Exercise 1d: Validate Passwordless Sign-in

1) Switch back to the in-private browser window sign in again by using the [user account](#).



alexw@m365x140515.onmicrosoft.com

### Meer informatie vereist

Uw organisatie heeft meer informatie nodig om uw account veilig te houden

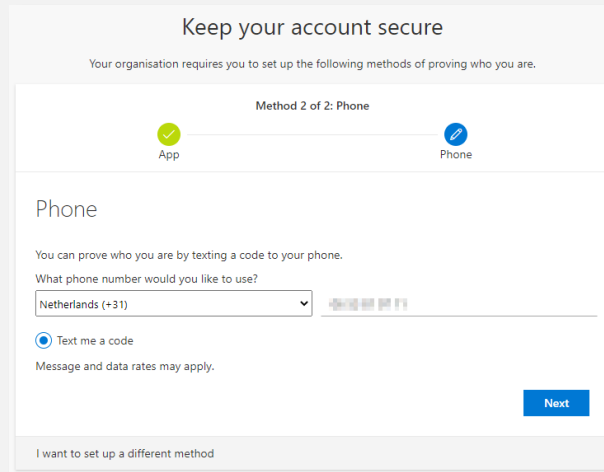
[Nu overslaan \(14 dagen totdat dit is vereist\)](#)

[Een ander account gebruiken](#)

[Meer informatie](#)

Volgende

- 2) 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)

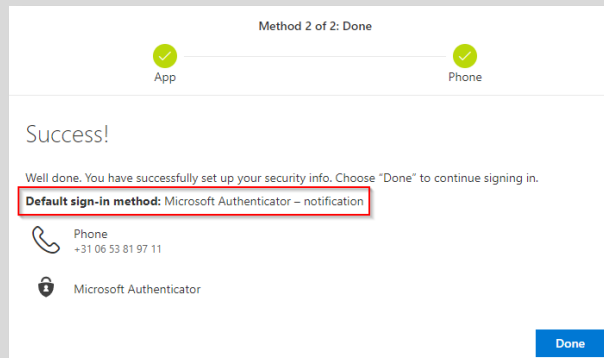
☒ Text me a code

Message and data rates may apply.

Next

I want to set up a different method

- 3) When done, the Authenticator should be the default sign-in method. The phone number is an alternative 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.

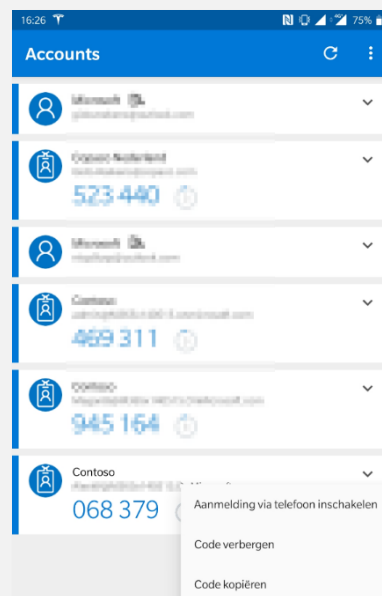
**Default sign-in method:** Microsoft Authenticator - notification

Phone  
+31 06 53 81 97 11

Microsoft Authenticator

Done

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



- 5) Sign in using the user credentials and register the device for Phone Sign-In.

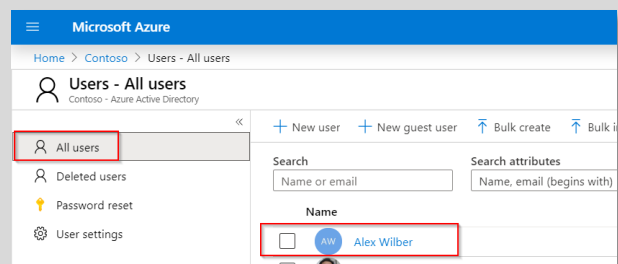
- 6) From the web browser, sign out from the account. Sign right back in using the same user account.



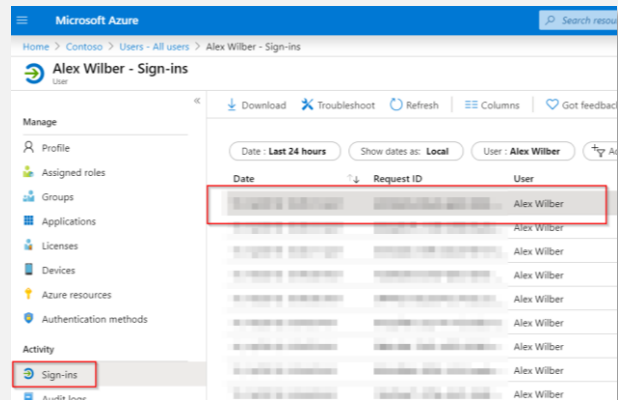
- 7) Please note you won't be prompted for a password. Instead, the Authenticator app now validates using a numeric value. Choose the corresponding number to validate.

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

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



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



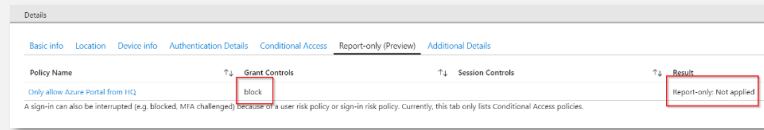
- 11) Browse to the Conditional Access tab. Validate if the policies are applied as expected. This helps you understand behavior and troubleshooting concerning sign-ins.

Details				
Policy Name	Grant Controls	Session Controls	Result	
Baseline policy: End user protection (Preview)				Success
Baseline policy: Require MFA for admins (Preview)	require multi-factor authentication			Not Applied
Baseline policy: Block legacy authentication (Preview)	block			Not Applied
Baseline policy: Require MFA for Service Management	require multi-factor authentication			Not Applied

A sign-in can also be interrupted (e.g. blocked, MFA challenged) because of a user risk policy or sign-in risk policy. Currently, this tab only lists Conditional Access policies.



12) 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.



13) Your done for this exercise!

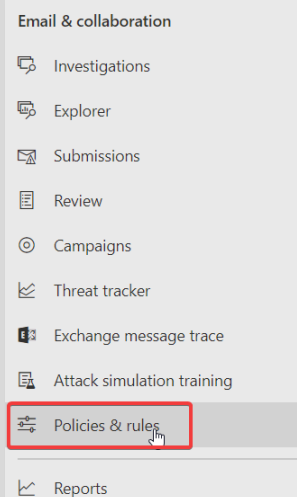
## Activity 2 : Prevent against phishing attacks with Defender for Office 365

*Estimated time to complete this activity*

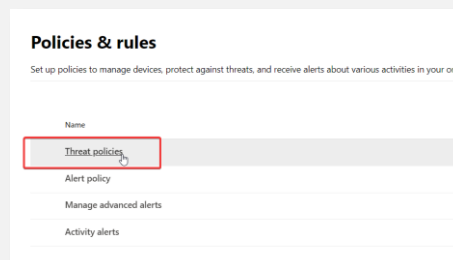
15 minutes

- 1) Go to <https://security.microsoft.com>

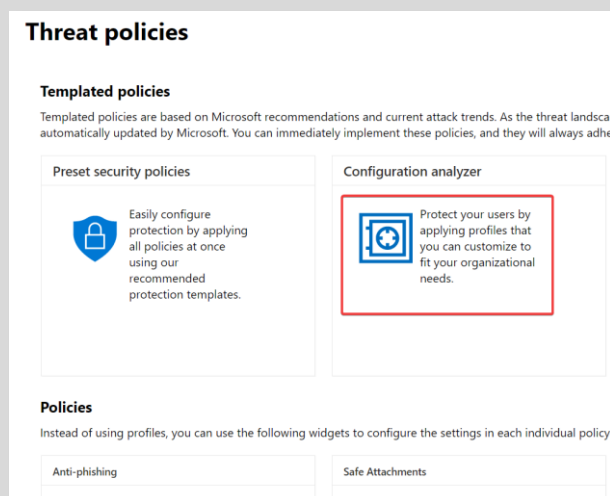
Browse to the **Policies & Rules** in the **Email & Collaboration** section



- 2) Open the **Threat Policies**



- 3) Here you can set policies concerning Exchange Online Protection and Defender for Office 365. We'll be using the **Configuration Analyzer** to finetune the default deployment.



- 4) Take note that the default configuration is missing some important settings and the indicator is showing a red flag. Users are not fully secured from spam, malware and phishing attacks.

We'll be focusing on the anti phishing policies in this lab, but feel free to analyze the other recommendations.

> Policy group/setting name	
> Anti-spam	5 recommendations
> Anti-phishing	11 recommendations
> Anti-malware	1 recommendations
> Safe Attachments	All settings follow Standard recommendations
> Safe Links	All settings follow Standard recommendations

- 5) Open the anti-phishing recommendations.

Anti-phishing 11 recommendations					
Add users to protect	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Modify
Automatically include the domains I own	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Adopt
Include custom domains	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Modify
If email is sent by an impersonated user	Office365 AntiPhish ...	No action	Jan 30, 2021 5:01 AM	Quarantine message	Adopt
If email is sent by an impersonated domain	Office365 AntiPhish ...	No action	Jan 30, 2021 5:01 AM	Quarantine message	Adopt
Enable impersonation protection that uses mailbox...	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Adopt
Show tip for impersonated users	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Adopt
Show tip for impersonated domains	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Adopt
Show tip for unusual characters	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	True	Adopt
If email is sent by an impersonated user who's prot...	Office365 AntiPhish ...	No action	Jan 30, 2021 5:01 AM	Move to Junk Email folder	Adopt
Advanced phishing thresholds	Office365 AntiPhish ...	1	Jan 30, 2021 5:01 AM	2	Adopt

- 6) Start by adopting the recommendations to show mailtips to users when phishing is suspected.

Show tip for impersonated users	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	Recommendations successfully adopted
Show tip for impersonated domains	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	Recommendations successfully adopted
Show tip for unusual characters	Office365 AntiPhish ...	False	Jan 30, 2021 5:01 AM	Recommendations successfully adopted

- 7) Make sure phishing messages are being delivered to Junk or Quarantine.

If email is sent by an impersonated user	Office365 AntiPhish ...	No action	Feb 16, 2021 1:52 PM	Recommendations successfully adopted
If email is sent by an impersonated domain	Office365 AntiPhish ...	No action	Feb 16, 2021 1:52 PM	Recommendations successfully adopted
Enable impersonation protection that uses mailbox...	Office365 AntiPhish ...	False	Feb 16, 2021 1:52 PM	True Adopt
If email is sent by an impersonated user who's prot...	Office365 AntiPhish ...	No action	Feb 16, 2021 1:52 PM	Recommendations successfully adopted

- 8) Include the domains you use for email delivery to enable protection.

Automatically include the domains I own	Office365 AntiPhish ...	False	Feb 16, 2021 1:54 PM	Recommendations successfully
---	-------------------------	-------	----------------------	------------------------------

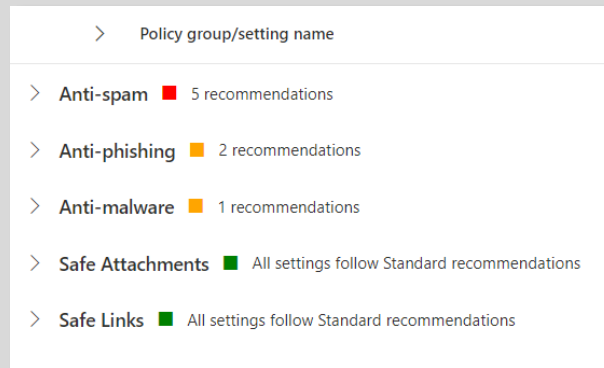
- 9) Adjust the threshold to the recommended values.

Enable impersonation protection that uses mailbox...	Office365 AntiPhish ...	False	Feb 16, 2021 1:58 PM	Recommendations successfully adopted
Advanced phishing thresholds	Office365 AntiPhish ...	1	Feb 16, 2021 1:58 PM	Recommendations successfully adopted

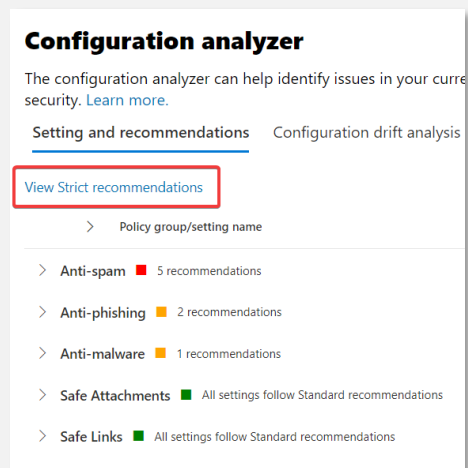
- 10) In a production environment, you should include any custom domains configured and

enable anti impersonation protection to vulnerable accounts. They'll be skipped in this lab.

11) Review the updated score. Users are now better protected against phishing attacks.



12) Change the recommendations to strict.



13) Review the adjusted recommendations. Please note additional changes are needed to comply with a strict anti-phishing policy. The red flag has returned to represent this.

Anti-phishing 5 recommendations				
If email is sent by someone who's not allowed to sp...	Office365 AntiPhish ...	Move to Junk Email f...	Feb 16, 2021 2:00 PM	Quarantine message Adopt
Add users to protect	Office365 AntiPhish ...	False	Feb 16, 2021 2:00 PM	True Modify
Include custom domains	Office365 AntiPhish ...	False	Feb 16, 2021 2:00 PM	True Modify
If email is sent by an impersonated user who's prot...	Office365 AntiPhish ...	Move to Junk Email f...	Feb 16, 2021 2:00 PM	Quarantine message Adopt
Advanced phishing thresholds	Office365 AntiPhish ...	2	Feb 16, 2021 2:00 PM	3 Adopt

14) This ends the current activity.

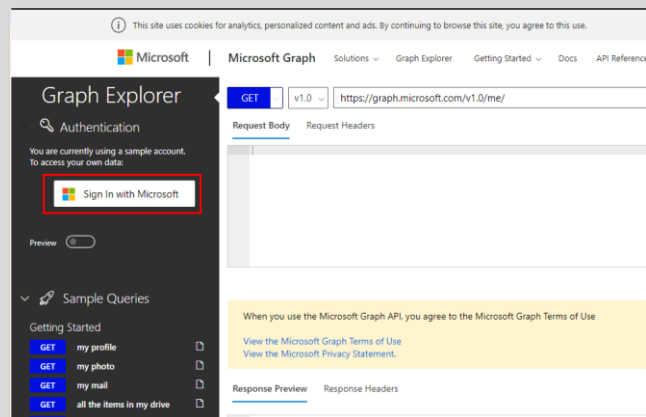
## Activity 3 : Graph API

*Estimated time to complete this activity*

45 minutes

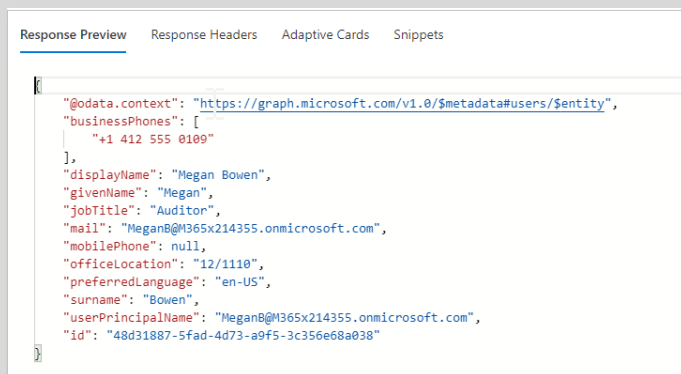
### Exercise 3a : Graph API Explorer

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



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

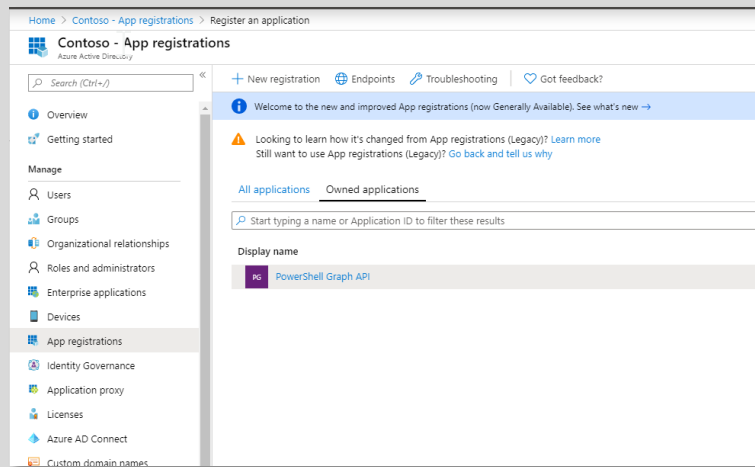
- 19) At the top of the screen, choose Run Query.
- 20) Review the output



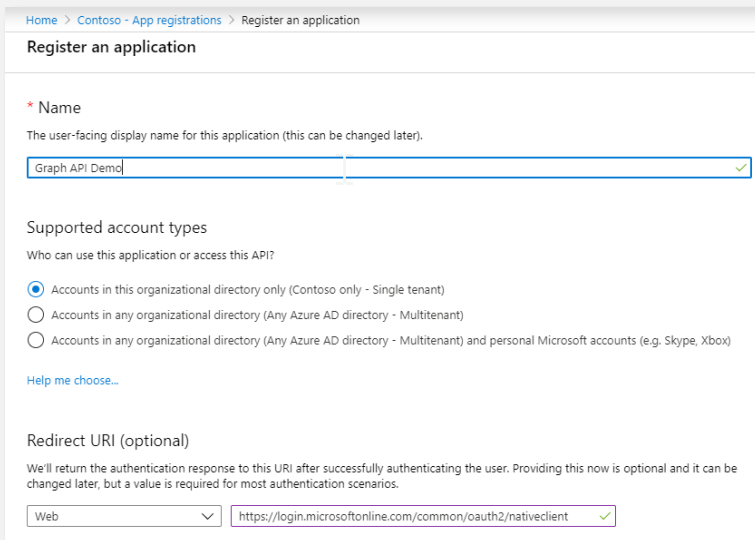
- 21) Also make note of the Permissions tab. This is an easy way to discover what permissions are required for the query.

## Exercise 3b : Connect with the Graph API

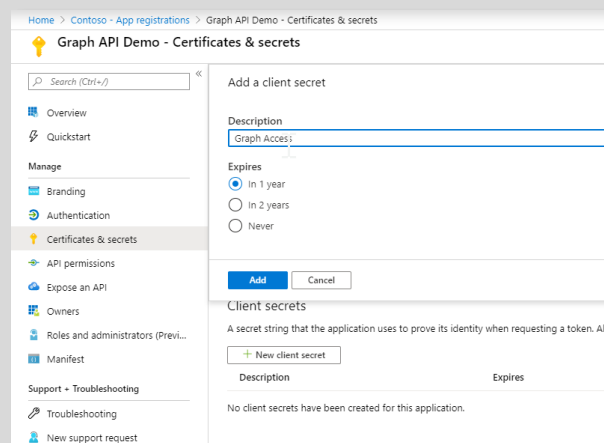
- 1) Log in to portal.azure.com
- 2) Go to Azure Active Directory
- 3) Choose App Registrations
- 4) Choose New Registration



- 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



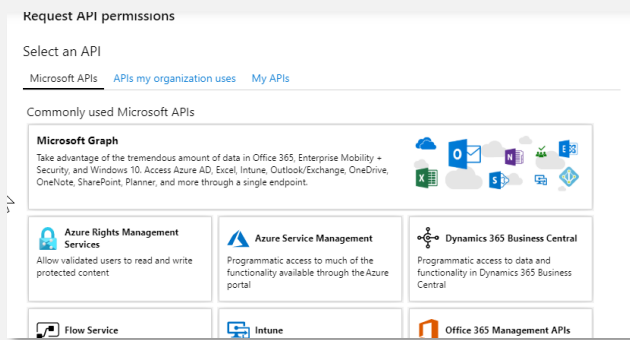
- 9) Go to Certificates & secrets and choose New Client Secret
- 10) Choose 1 Year and give it an Description
- 11) Choose Add
- 12) **Copy the Client Secret value** you wont be able to get it after you leave the page. And store it some where save



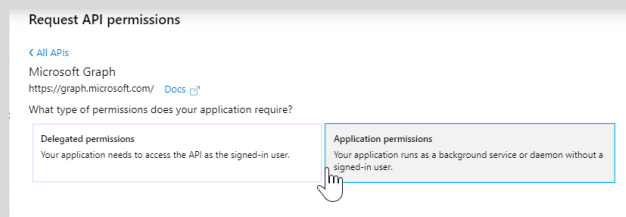
13) Go to API permissions

14) Choose Add a permission

15) Select Microsoft Graph



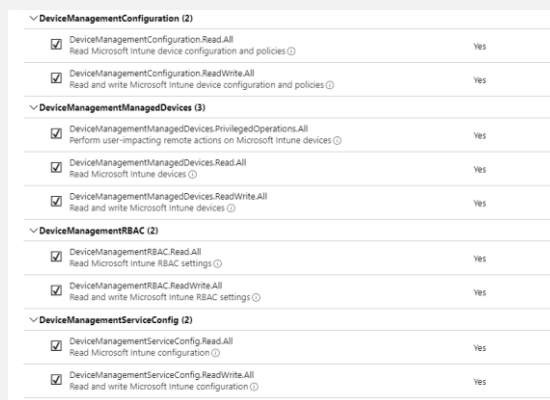
16) Choose Application permissions  
(for the demo this is easiest)



17) You can now select which permissions are needed.

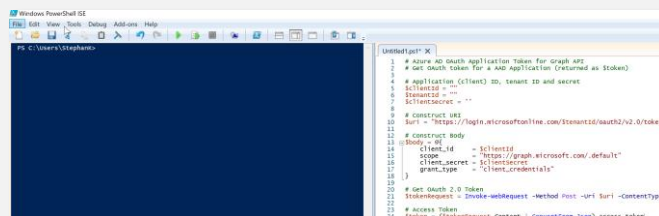
18) Select all user and devices permissions and choose add permissions

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> and select the app that you have registered. In the overview





## Exercise 3c : 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 + **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!**

- 3) Get all users

```
#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
```

- 4) Get specific user
- 5) Make sure that you adjust the upn for the specific

```
$users.value | Select-Object DisplayName, ID, UserPrincipalName

#get specific user
$uri = "https://graph.microsoft.com/beta/users/CameronW@M3"
$megan = invoke-restmethod -Method GET -Uri $uri -Headers
$megan.displayName
$megan | Select-Object DisplayName, MobilePhone, City
```

- 6) Update user info

```
#update user info
$PatchJSON = @{"mobilephone" = "+31640409642"
               "city" = "Eindhoven"
               } | ConvertTo-Json

Invoke-RestMethod -Uri $uri -Method PATCH -Headers @{Auth
```

- 7) Check updated info
- 8) Make sure that you adjust the upn for the specific

```
#check if user info is updated
$uri = "https://graph.microsoft.com/beta/users/CameronW@M3"
$megan = invoke-restmethod -Method GET -Uri $uri -Headers
$megan.displayName
$megan | Select-Object DisplayName, MobilePhone, City
```

- 9) Create new user
- 10) Make sure that you adjust the upn your tenant

```
#create new user
$uri = "https://graph.microsoft.com/beta/users"
$NewUserJSON = @{"accountEnabled" = $true
                 "displayName" = "EL Demo User"
                 "mailNickname" = "elDEMouser"
                 "userPrincipalName" = "elDEMouser@M365x428595.0"
                 "mobilephone" = "+31640409642"
                 "city" = "Eindhoven"
                 "passwordProfile" = @{"forceChangePasswordNextSignIn" = $true
```

- 11) Check created user

```
#check created user
$uri = "https://graph.microsoft.com/beta/user
$DemoUser = invoke-restmethod -Method GET -UR
$DemoUser.displayName
$DemoUser | Select-Object DisplayName, Mobile
```



12) Delete created user

```
#get and delete created user
$uri = "https://graph.microsoft.com/beta/users"
$uri = $uri + '/' + $response.id
invoke-restmethod -Method GET -Uri $uri -Headers @
Invoke-RestMethod -Method DELETE -Uri $uri -Header:
```

13) Get all groups

```
#get all groups
$uri = "https://graph.microsoft.com/beta/groups"
Invoke-RestMethod -Method GET -Uri $uri -Header
$groups = Invoke-RestMethod -Method GET -Uri
$groups.value
$groups.value | ft DisplayName
```

14) Get Member of first group

```
#get member of first group
$groups.value[0]
$groupid = $groups.value[0].id
$uri = $uri + '/' + $groupid
Invoke-RestMethod -Method GET -Uri $uri -Header
$uri = $uri + '/' + 'members'
$members = Invoke-RestMethod -Method GET -Uri $
```

15) Get groups a user is member of

```
#get groups user is member of
$uri = "https://graph.microsoft.com/beta/users"
$membership = Invoke-RestMethod -Method GET -Uri
$membership.value | select DisplayName
```

16) Export Device configuration Profiles

```
1
2 ###Export Device configuration profiles
3 $uri = "https://graph.microsoft.com/beta/deviceManagement/configurationProfiles"
4 $configs = Invoke-RestMethod -Method GET -Uri $uri
5
6 foreach ($config in $configs.value) {
7     $configname = $config.displayName
8     $configfile = "C:\temp\$configname" + '.json'
9     $config | ConvertTo-Json | out-file $configfile
10 }
11
```

17) Export Device compliance policies

```
###Export Device Compliance Policies
$uri = "https://graph.microsoft.com/beta/deviceManagement/compliancePolicies"
$compliances = Invoke-RestMethod -Method GET -Uri

foreach ($compliance in $compliances.value) {
    $configname = $compliance.displayName
    $configfile = "C:\temp\$configname" + '.json'
    $config | ConvertTo-Json | out-file $configfile
}
```

18) Hello For Business settings

```
###Export Hello For Business Settings
$uri = "https://graph.microsoft.com/beta/deviceManage
$WHBusiness = Invoke-RestMethod -Method GET -Uri $uri
$WHBusiness

$configname = $WHBusiness.value | select Displayname
$configfile = "C:\temp\${$configname[0]}" + '.json'
$config | ConvertTo-Json | out-file $configfile
```

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.

```
#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
$outputNieuwPolicy = Invoke-RestMethod -Method POST -Uri $uri -Headers @{Authorization = "
$uri = "https://graph.microsoft.com/beta/deviceManagement/deviceconfigurations/${$outputNieuwPolicy}
Invoke-RestMethod -Method GET -Uri $uri -Headers @{Authorization = "Bearer $token" } -Error
```

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

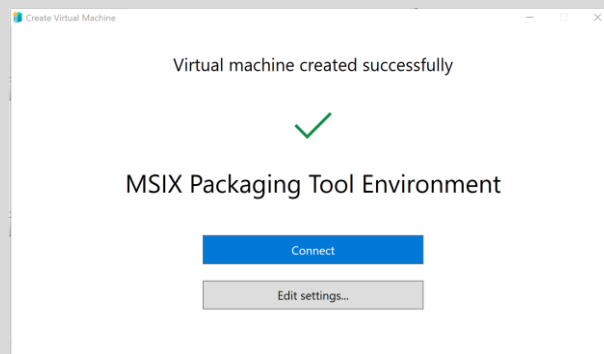
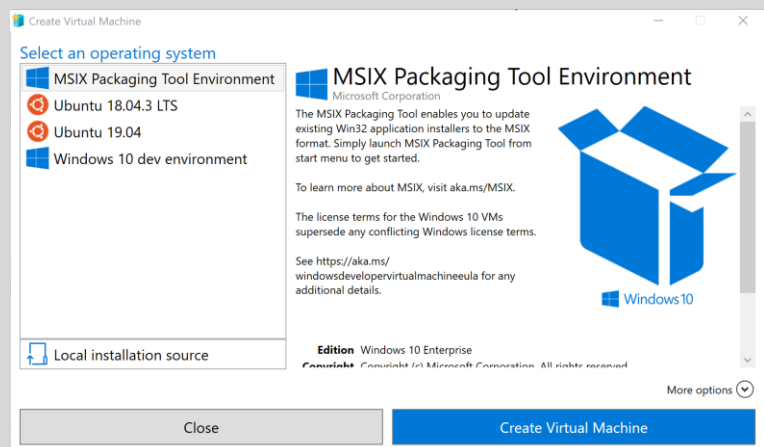
## Activity 4 : MSIX

*Estimated time to complete this activity*

60 minutes

### Exercise 4a : Create Packaging VM

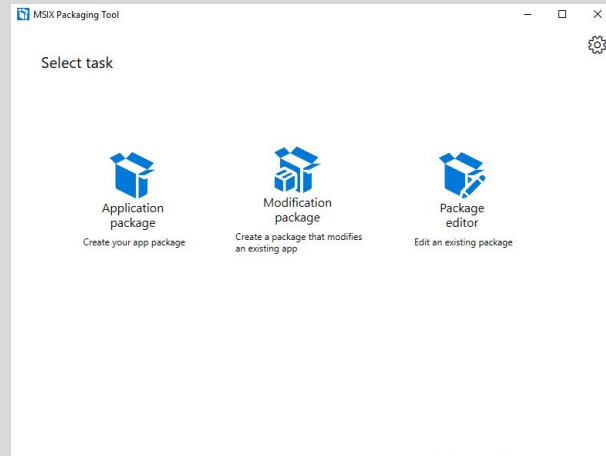
- 1) Open Hyper-V manager
- 2) Right click your host name and choose Quick Create
- 3) Pick MSIX Packaging Tool Environment and choose **Create Virtual Machine**. This will start to download the VM for you
- 4) When the VM is ready you can connect to the VM



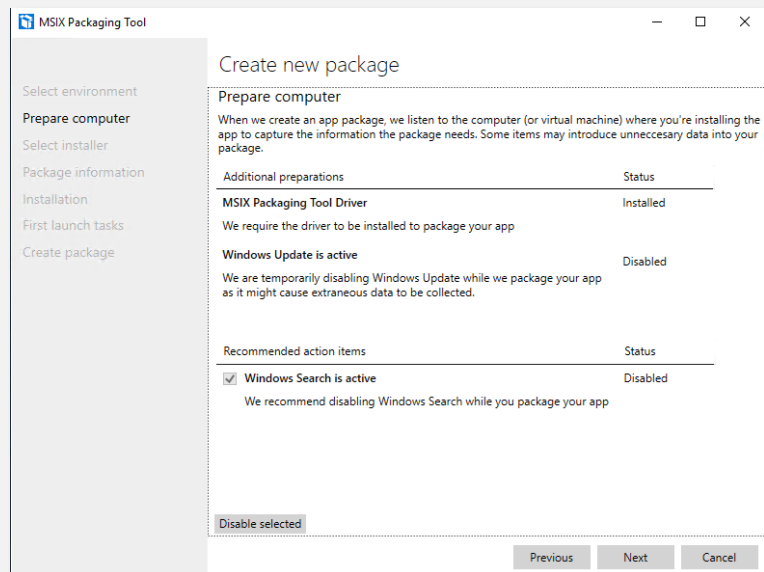
- 5) Start the VM and go through the installation process. Choose to domain join instead of a Microsoft account
- 6) When the VM is ready to use, it best that you create a snapshot (check point) so you can revert to this clean state in the future.
- 7) Download the Citrix receiver app and the pfx certificate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

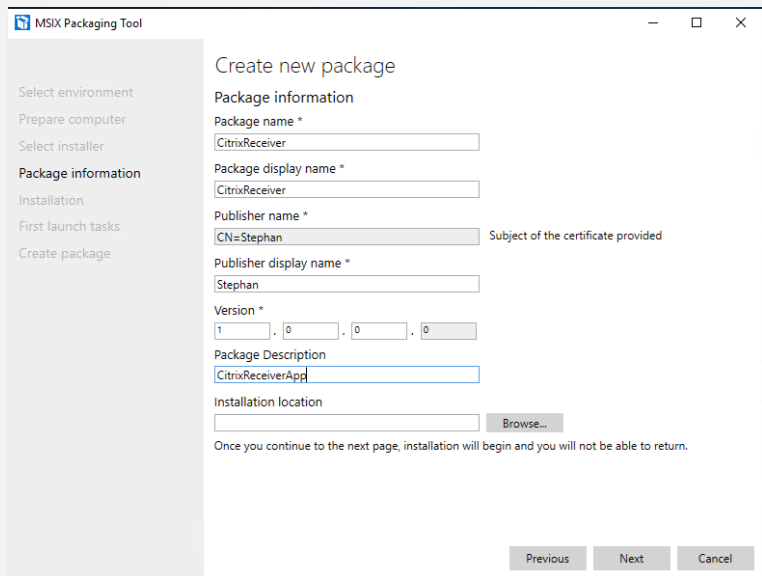


- 11) Create Package on this computer and click next.
- 12) Disable Windows Search and click next.



- 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

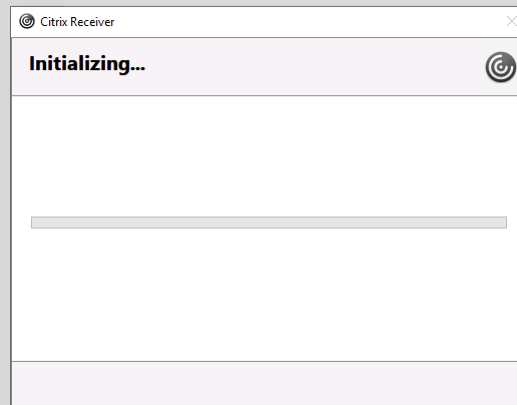


The screenshot shows the 'MSIX Packaging Tool' window with the 'Create new package' tab selected. The 'Package information' section contains the following fields:

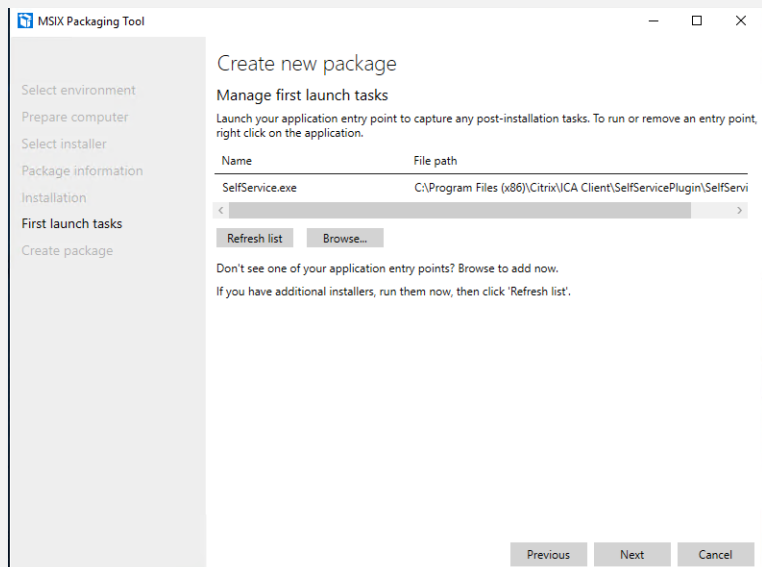
- Package name \*: CitrixReceiver
- Package display name \*: CitrixReceiver
- Publisher name \*: CN=Stephan (Subject of the certificate provided)
- Publisher display name \*: Stephan
- Version \*: 1.0.0.0
- Package Description: CitrixReceiverApp
- Installation location: (empty) [Browse...]

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

- 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



- 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 'Manage first launch tasks' tab selected. The 'Manage first launch tasks' section contains the following information:

- Launch your application entry point to capture any post-installation tasks. To run or remove an entry point, right click on the application.
- Table with columns 'Name' and 'File path':
 

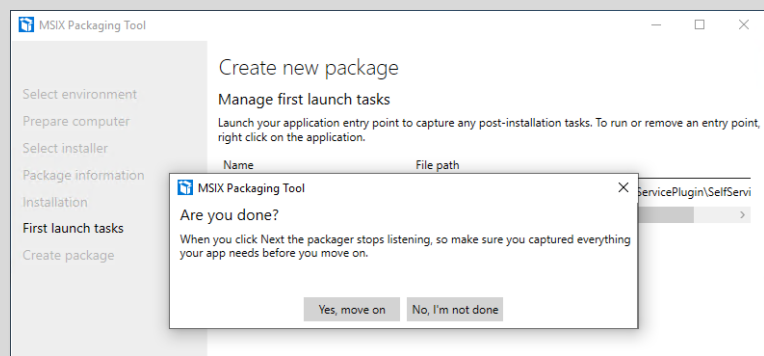
Name	File path
SelfService.exe	C:\Program Files (x86)\Citrix\ICA Client\SelfServicePlugin\SelfServi
- Buttons: Refresh list, Browse...
- Text: Don't see one of your application entry points? Browse to add now. If you have additional installers, run them now, then click 'Refresh list'.

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

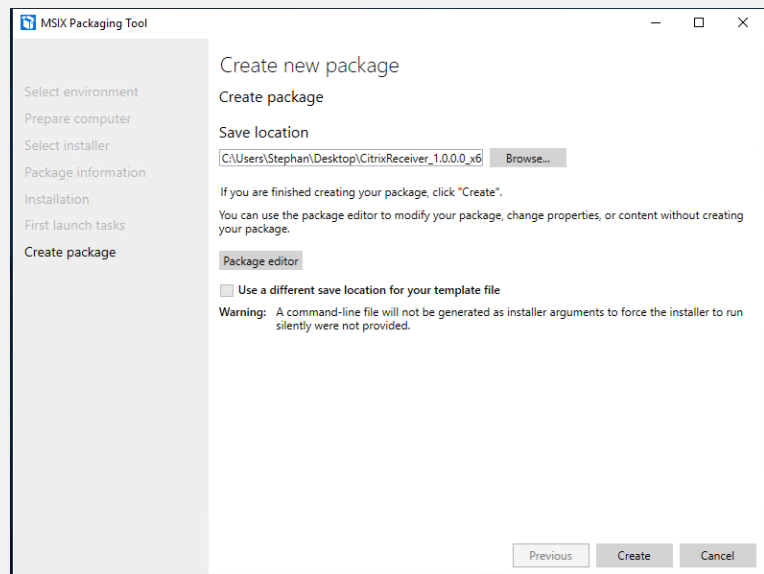


23) Yes, move on

24) Your package is created



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



## Exercise 4b : Test your package

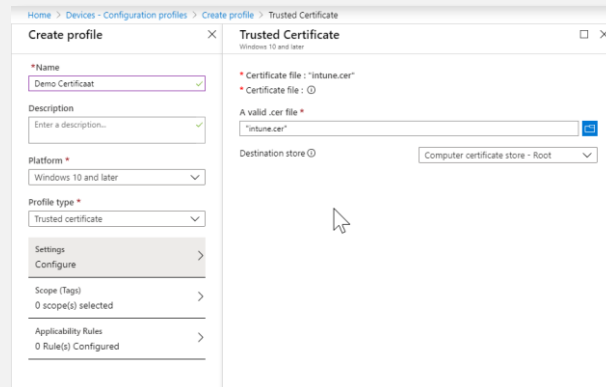
Now you created the package you probably want to test it. There are a couple of ways you can do this. The first (and most easy way to do this) is to use the same MSIX machine you created. Since this being an M365 lab we can also deploy the MSIX with Intune, but this requires you to have a test VM which 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.

## Exercise 4c : 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

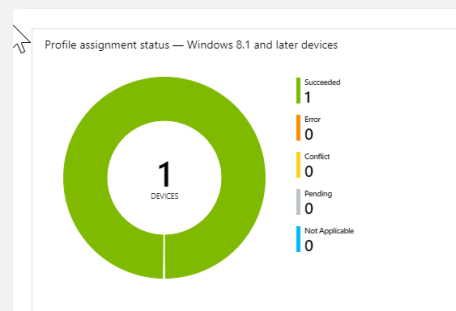
- 1) Go to <https://devicemanagement.microsoft.com/> and to Devices. Here you can create a new Configuration Profile

- 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



- 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

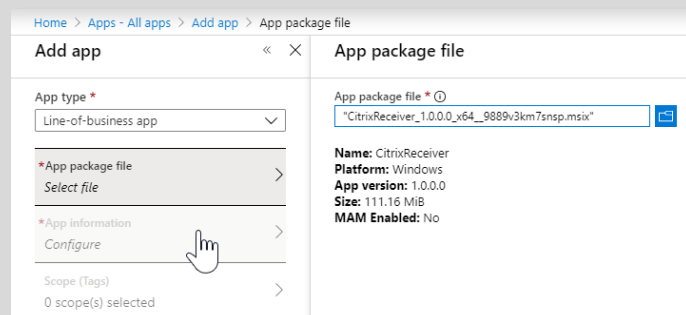
- 11) In the device management portal go to **Apps → All apps** and choose **Add**

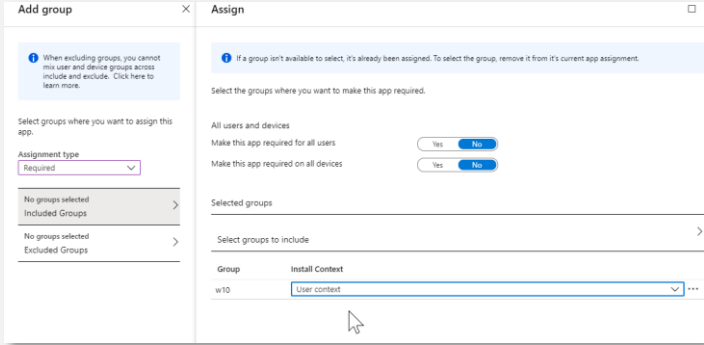
- 12) App type is **Line-of-Business app**

- 13) At App package file navigate to your Citrix receiver MSIX package

- 14) Choose OK

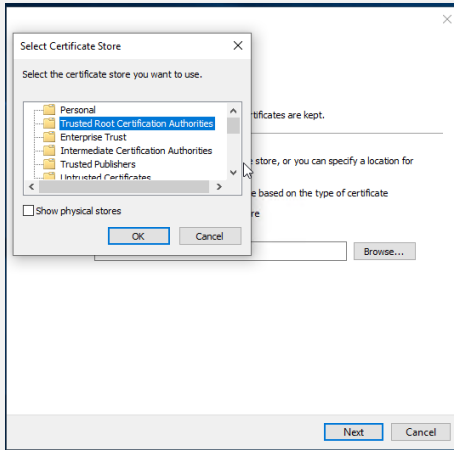
- 15) Open the App information settings and reviews the settings. When ready choose



OK	
16) Choose Add	
17) The application is now being uploaded. When ready you can continue to the next step	
18) Go to assignment and choose <b>Add Group</b>	
19) For assignment type choose <b>Required</b>	
20) Pick the group which has your test vm in it and install in the <b>User Context</b>	
21) Click Save	
22) Wait for the deployment to finish	

### Exercise 4d : Test the Package locally

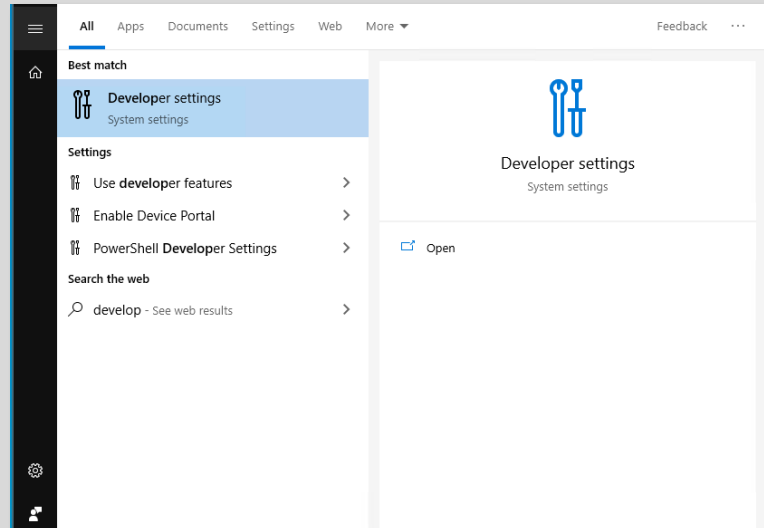
If you don't have a test VM which is enrolled in Intune you can also use the MSIX packaging machine to test your package.

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 <b>Welkom123\$</b> , and choose next	
8) Choose Place all certificates in the following store	

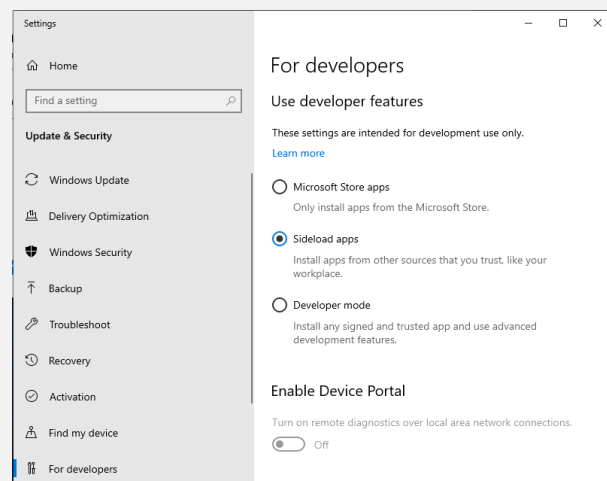
9) Select **Trusted Root Certification Authorities** Next

10) Finish

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

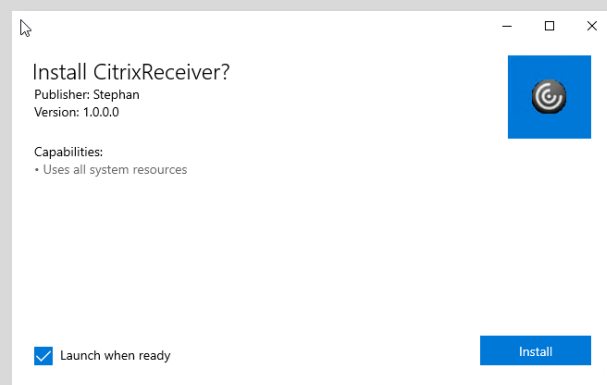


12) Choose Sideload apps

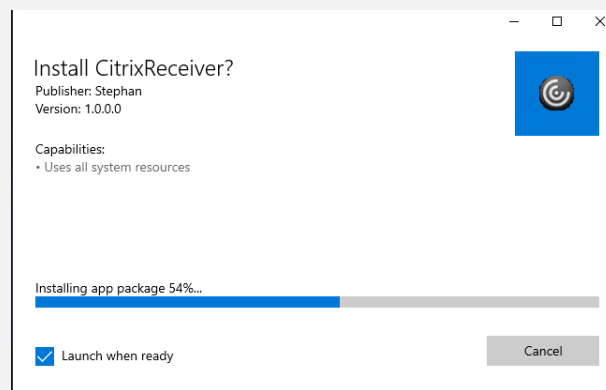


13) Then Double click your MSIX package

14) Choose Install



15) The applications is being installed



16)

