

2021

Conditional Access demystified whitepaper

WHITEPAPER ON IMPLEMENTING AZURE ACTIVE DIRECTORY
CONDITIONAL ACCESS

KENNETH VAN SURKSUM | WWW.VANSURKSUM.COM | INSIGHT24
VERSION 1.3 | OCTOBER 2021





Contents

1	Introduction	7
1.1	Why this whitepaper?.....	7
2	What is Conditional Access?.....	8
2.1	Licensing.....	9
2.2	Security defaults.....	9
2.3	Secure score	10
2.4	What's up with the preview label?.....	11
2.5	Legacy authentication.....	12
3	How does Conditional Access work?.....	13
3.1	Custom Conditional Access policies	14
3.2	Structure of a Conditional Access policy explained.....	17
3.2.1	General.....	17
3.2.2	Assignments	18
3.2.2.1	Users and Groups.....	18
3.2.2.2	Cloud apps, user actions or authentication context.....	18
3.2.2.3	Conditions.....	19
3.2.2.3.1	User Risk (additional license needed)	19
3.2.2.3.2	Sign-in Risk (additional license needed).....	19
3.2.2.3.3	Device Platforms.....	20
3.2.2.3.4	Locations.....	20
3.2.2.3.5	Client apps	21
3.2.2.3.6	Device state	21
3.2.2.3.7	Filter for devices.....	21
3.2.2.3.7.1	Create include or exclude filter.....	22
3.2.3	Access Controls.....	25
3.2.3.1	Grant	25
3.2.3.2	Session Controls.....	26
3.2.3.3	App enforced restrictions.....	27
3.2.3.4	Control Azure AD Conditional Access policy behavior during an Azure AD outage ..	27
4	Designing a Conditional Access strategy	29
4.1	Service dependencies	30
4.2	Functional Design	31
4.2.1	Guest users.....	31
4.2.2	Allow full access to the environment.....	32
4.2.3	Allow Browser/Browser restricted access to the environment.	32
4.2.4	Device Owner and usage.....	32
4.2.4.1	Company owned.....	32





4.2.4.2	Personally owned.....	32
4.2.4.3	Company owned, personally used.....	33
4.2.4.4	Other Company owned	33
4.3	Browser restrictions and configuration when using Conditional Access.....	34
4.3.1	Browser support	34
4.3.1.1	Sign-in Logging.....	34
4.3.2	Mozilla FireFox.....	35
4.3.2.1	Configuring the settings from Microsoft Endpoint Manager	35
4.3.2.2	Create the custom Device Configuration Profile	36
4.3.3	Google Chrome.....	36
4.3.3.1	Deploying extensions for Google Chrome using Microsoft Endpoint Manager	37
4.3.4	Microsoft Edge.....	39
4.4	Azure AD Conditional Access authentication context	41
4.4.1	Current data protection options.....	41
4.4.2	Conditional Access App Enforced Restrictions	42
4.4.3	Microsoft Cloud App Security (MCAS)	42
4.4.4	Conditional Access Authentication Context.....	42
4.4.5	Create Authentication context labels	43
4.4.6	Use the Authentication context label in your sensitivity labels	44
4.4.7	Creating the Conditional Access policy for the Authentication context label.....	44
4.4.8	Impact on using App enforced restrictions	46
4.4.9	Use the Authentication context label in your MCAS Session Policy.....	47
4.5	Understanding and governing reauthentication settings in Azure Active Directory	48
4.5.1	The settings which make up the experience.....	48
4.5.1.1	Azure Multi Factor Authentication Settings	48
4.5.1.2	Show option to remain signed in (KMSI).....	49
4.5.1.3	Conditional Access	49
4.5.1.4	Sign-in Frequency.....	50
4.5.1.5	Persistent Browser session.....	50
4.5.2	The scenarios which make up the experience.....	50
4.5.2.1	Supported applications	51
4.5.2.2	Nonregistered devices.....	51
4.5.2.3	Azure AD joined or Registered devices	51
4.5.3	Managed devices	51
4.5.4	Browser used.....	52
4.5.5	Default settings when creating a new tenant	52
4.5.5.1	Multi factor authentication	52
4.5.5.2	Show option to remain signed in.....	52





4.5.5.3	Conditional Access	53
4.5.6	Bringing it all together	53
4.5.6.1	Managed devices	53
4.5.6.2	Managed applications.....	53
4.5.6.3	Non-managed devices.....	54
5	Implementing Conditional Access.....	55
5.1	My recommended default set of policies	59
5.1.1	Versioning.....	60
5.1.2	Prerequisites.....	60
5.1.2.1	CAP001-All: Block Legacy Authentication for All users when OtherClients-v1.0.....	60
5.1.2.2	CAP002-O365: Grant Exchange ActiveSync Clients for All users when Approved App-v1.0	60
5.1.3	User	61
5.1.3.1	CAU001-All: Grant Require MFA for guests when Browser and Modern Auth Clients-v1.0	61
5.1.3.2	CAU002-All: Grant Require MFA for All users when Browser and Modern Auth Clients-v1.0	61
5.1.3.3	CAU003-Selected: Block unapproved apps for guests when Browser and Modern Auth Clients-v1.0.....	61
5.1.3.4	CAU004-Selected: Session route through MCAS for All users when Browser on Non-Compliant-v1.1.....	61
5.1.3.5	CAU005-Selected: Session route through MCAS for All users when Browser on Compliant-v1.0.....	62
5.1.3.6	CAU006-All: Grant access for High Risk Sign-in for All Users when Browser and Modern Auth Clients require MFA-v1.0	62
5.1.3.7	CAU007-All: Grant access for High-Risk Users for All Users when Browser and Modern Auth Clients require PWD reset-v1.0	62
5.1.3.8	CAU008-All: Grant Require MFA for Admins when Browser and Modern Auth Clients-v1.0	62
5.1.3.9	CAU009-AzureManagement: Grant Require MFA for Azure Management for All Users when Browser and Modern Auth Clients-v1.0	63
5.1.3.10	CAU010-All: Grant Require ToU for All Users when Browser and Modern Auth Clients-v1.0 (Optional).....	63
5.1.3.11	CAU011-All: Block access for All users except licensed when Browser and Modern Auth Clients-v1.0 (Optional)	63
5.1.4	Device.....	64
5.1.4.1	CAD001-O365: Grant macOS access for All users when Browser and Modern Auth Clients and Compliant-v1.0.....	64
5.1.4.2	CAD002-O365: Grant Windows access for All users when Browser and Modern Auth Clients and Compliant-v1.0.....	64
5.1.4.3	CAD003-O365: Grant iOS and Android access for All users when Modern Auth Clients and ApprovedApp and Compliant-v1.0.....	64





5.1.4.4 CAD004-O365: Grant Require MFA for All users when Browser and Non-Compliant-v1.1	65
5.1.4.5 CAD005-O365: Block access for unsupported device platforms for All users when Modern Auth Clients-v1.0	65
5.1.4.6 CAD006-O365: Session block download on unmanaged device when All users when Browser-v1.1.....	66
5.1.4.7 CAD007-O365: Session set Sign-in Frequency for Apps for All users when Modern Auth Clients and Non-Compliant-v1.0.....	66
5.1.4.8 CAD008-All: Session set Sign-in Frequency for All users when Browser and Non-Compliant-v1.1.....	66
5.1.4.9 CAD009-All: Session disable browser persistence for All users when Browser and Non-Compliant-v1.1	67
5.1.4.10 CAD010-All: Require MFA for device join or registration when Browser and Modern Auth Clients-v1.0	67
5.1.5 Location.....	67
5.1.5.1 CAL001-All: Block specified locations for All users when Browser and Modern Auth Clients-v1.0	67
5.1.5.2 CAL002:All: Require MFA registration from trusted locations only for All users when Browser and Modern Auth Clients-v1.0	67
5.1.6 Translating the functional design to the technical implementation	68
5.2 Limit Access to Outlook Web Access and SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions.....	69
5.2.1 Configure Outlook Web Access for limited access via App Enforced Restrictions	69
5.2.2 Configure SharePoint Online and OneDrive for limited access via App Enforced Restrictions	70
5.2.2.1 Global settings	70
5.2.2.2 Per site settings.....	71
5.2.2.3 Using sensitivity labels for containers.....	72
5.2.2.3.1 Option 1: Block or limit access to a specific SharePoint site or OneDrive.....	72
5.2.2.3.2 Option 2: Use Sensitivity labels for Containers.....	73
5.2.2.3.3 How to provide a Sensitivity label to your already existing Teams and SharePoint sites? 82	
5.2.2.4 Policy behavior within Outlook Web Access	84
5.2.2.5 Policy behavior within SharePoint Online and OneDrive.....	84
6 Testing and troubleshooting conditional access.....	86
6.1 What if tool	86
6.2 Report-only Mode	87
6.3 Azure Active Directory sign-ins logging	88
6.4 OneDrive client sign-in issues due to Conditional Access policies in Azure AD tenant where you are a guest user.....	89
6.4.1 So, what was going on?.....	90





6.4.2	Lessons learned	90
6.5	Where to find help and provide feedback.....	90
7	Modifying Conditional Access to suit your special needs.	91
8	Resources and further references	93
8.1	Microsoft documentation.....	93
8.2	Other interesting blogs.....	94





About: Kenneth van Surksum

Kenneth van Surksum works as a modern workplace consultant at [Insight24](#) and is specialized in building modern workplace solutions on top of Microsoft 365. Kenneth also works with public and private cloud solutions based on Azure and System Center. Kenneth is a [Microsoft MVP](#) for Enterprise Mobility.



Kenneth is co-founder of the [Workplace Ninja User Group Netherlands](#) and organizes (virtual) community meetings on a regular basis.

Kenneth loves to speak about technical topics related to his daily work. Kenneth is Microsoft Certified Trainer (MCT) and has multiple certifications, he has received the MVP and VMware vExpert award multiple times.

A special thanks to Peter Daalmans for reviewing the content.

Changelog:

Name & Version	Released	Comment
Conditional Access demystified-v1.0	August 2019	Initial version based on my series of blogposts on this topic
Conditional Access demystified-v1.1	May 2020	Updated cheatsheet, implementation workflow, documentation spreadsheet,
Conditional Access demystified-v1.2	February 2021	Big update, going from 30 to 77 pages including updated reference sheets. Describing some more scenario's and providing a set of default conditional access policies for anyone to use as a starting point.
Conditional Access demystified -v1.3	October 2021	Updated content and integration of several Conditional Access related blog articles

If you have any suggestions or feedback, please reach out to me via:

Email: kenneth@vansurksum.com

Twitter: [@kennethvhs](#)

LinkedIn: <https://www.linkedin.com/in/kennethvansurksum/>

Disclaimer: This information is provided "AS IS" with no warranties, confers no rights and is not supported by the author.

Copyright © 2021 by Kenneth van Surksum. All rights reserved. No part of the information on this web site may be reproduced or posted in any form or by any means without the prior written permission of the publisher.





1 Introduction

In July 2016 Microsoft made [Conditional Access generally available](#) as a feature of Azure Active Directory (AzureAD). Since that time, I had a love and hate relationship with this functionality of Azure AD. Mainly because it is difficult to test scenarios and some changes can have a really high impact. I even experienced being locked out of accessing the Azure portal during one of my tests.

1.1 Why this whitepaper?

There is already some good documentation from Microsoft and many blogposts by fellow bloggers detailing Conditional Access scenarios, but not really a one-stop shopping overview. With this whitepaper I hope to achieve this.

The paper combines several blogposts I wrote on this subject, where needed I will make references to those blogposts. I did my best to provide some structure in this paper, but it might be that you recognize that it exists of multiple blogposts tied together. I will do my upper best to correct that in future versions of this paper.

I will try to describe everything that I find important, and lessons learned while implementing Conditional Access in my own tenants and at customers. I will not go into much detail on creating individual Conditional Access policies, since that is both well documented by Microsoft and described by well-known bloggers on this subject like Peter van der Woude, Per Larsen, Alex Fields, Daniel Chronlund, Peter Daalmans, among others.

Microsoft is continuously adding functionality to Conditional Access, first functionality is added in a preview from which can be recognized by the (preview) tag in the name of the feature or Conditional Access policy and later it will eventually be released. The best way to keep up to date is by monitoring the [Azure Updates webpage](#), where available, in preview and in development features of Azure Active Directory are shared.



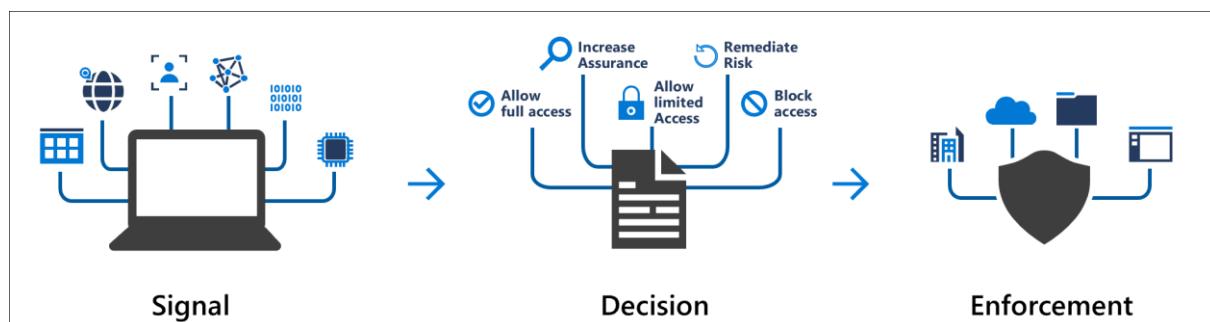


2 What is Conditional Access?

Organizations are moving services which they traditionally hosted in their on-premises environment to the cloud. Because of this traditional network security in the form of Firewalls and other equipment no longer offers full protection when it comes to protecting company data.

Identity has become the new perimeter and is becoming the new attack surface for bad actors. So besides protecting our traditional assets which are either hosted on-premises or within an IaaS environment we also must protect the identity. Azure AD Conditional Access is one of the available methods to protect your identity.

[Microsoft describes Conditional Access](#) as following: "*With Conditional Access, you can implement automated access control decisions for accessing your cloud apps that are based on conditions.*" and "*Conditional Access policies are enforced **after** the first-factor authentication has been completed. Therefore, Conditional Access is not intended as a first line defense for scenarios like denial-of-service (DoS) attacks, but can utilize signals from these events (e.g. the sign-in risk level, location of the request, and so on) to determine access.*"



The way I see it, the best way to explain what Conditional Access does, is by making the comparison to a firewall. A firewall determines what traffic can access your resources, under what circumstances and Conditional Access sort of does the same. Conditional Access describes under what circumstances users can access (by acquiring an Access Token) your cloud data or applications. Keep in mind though that Conditional Access policies are enforced **after** the first authentication has taken place.

With cloud applications in Azure AD, Microsoft references to applications which use Azure Active Directory (Azure AD) for authentication (and sometimes also for authorization). Azure Active Directory (Azure AD) is Microsoft's cloud-based identity and access management service. So, if you access a SaaS based application through Azure AD you can use Conditional Access as your "access firewall". Microsoft's SaaS offerings like Office 365, Dynamics, the Azure Portal all use Azure AD as its authentication and authorization mechanism.

This also means that if you can access the SaaS application in some other way, you can bypass Conditional Access making the solution less effective. For example, you store only the credentials of the application in Azure AD, allowing you to use SSO to login to the applications and making use of Conditional Access, but when accessing the SaaS application directly you can also provide your userid and password, which allows you to bypass the Conditional Access policies. So, it is important that you modify the SaaS identity provider to only allow Azure Active Directory signins, and not accept local logins anymore (except for maybe a breakglass account in case of emergency).





If you have SaaS apps which have a federation with your ADFS infrastructure, you can use claim rules in ADFS which provide similar functionality as Conditional Access, if you want to make use of Conditional Access you need to modify the federation to use Azure AD instead of ADFS.

2.1 Licensing

Conditional Access is a feature which is part of an [Azure AD Premium P1 and P2 license](#) which you can buy individually or is part of a suite license like Enterprise Mobility and Security (EM+S) E3/E5 or Microsoft 365 E3/E5, and [recently announced](#) Conditional Access is now also included as part of [Microsoft 365 Business Premium](#) licensing as well, since that license includes Azure AD Premium P1.

Some of the Conditional Access settings require that you have licensed other products, for example in order to [use the sign-in risk condition](#) you need to have Azure AD Identity Protection [licensed](#). (Part of Azure AD premium P2). In order integrate Conditional Access with Microsoft Cloud App Security (MCAS), you must have MCAS licensed as well.

Keep in mind, that you can apply Conditional Access policies to users which are not licensed for Azure AD Premium P1 and P2, since assigning Azure AD Premium P1/P2 to any user in the Azure AD puts the whole Azure AD in that modus. Even though this technically works, you are in conflict with the licensing terms. Microsoft states that ***if you use/benefit from a specific service within Azure, you must be licensed for it.***

Therefore, I advise to use group-based licensing and make sure that these groups are used for conditional access configuration as well.

2.2 Security defaults

If you are not licensed to use Azure AD Conditional Access, you can enable Security Defaults on your Azure AD tenant.

Microsoft explains the security defaults as following: “*Security defaults provide secure default settings that we manage on behalf of organizations to keep customers safe until they are ready to manage their own identity security story.*” For now, when the security defaults are enabled the following security settings are enforced:

1. Requiring all users and admins to register for MFA.
2. Challenging users with MFA – mostly when they show up on a new device or app, but more often for critical roles and tasks.
3. Disabling authentication from legacy authentication clients, which can't do MFA.

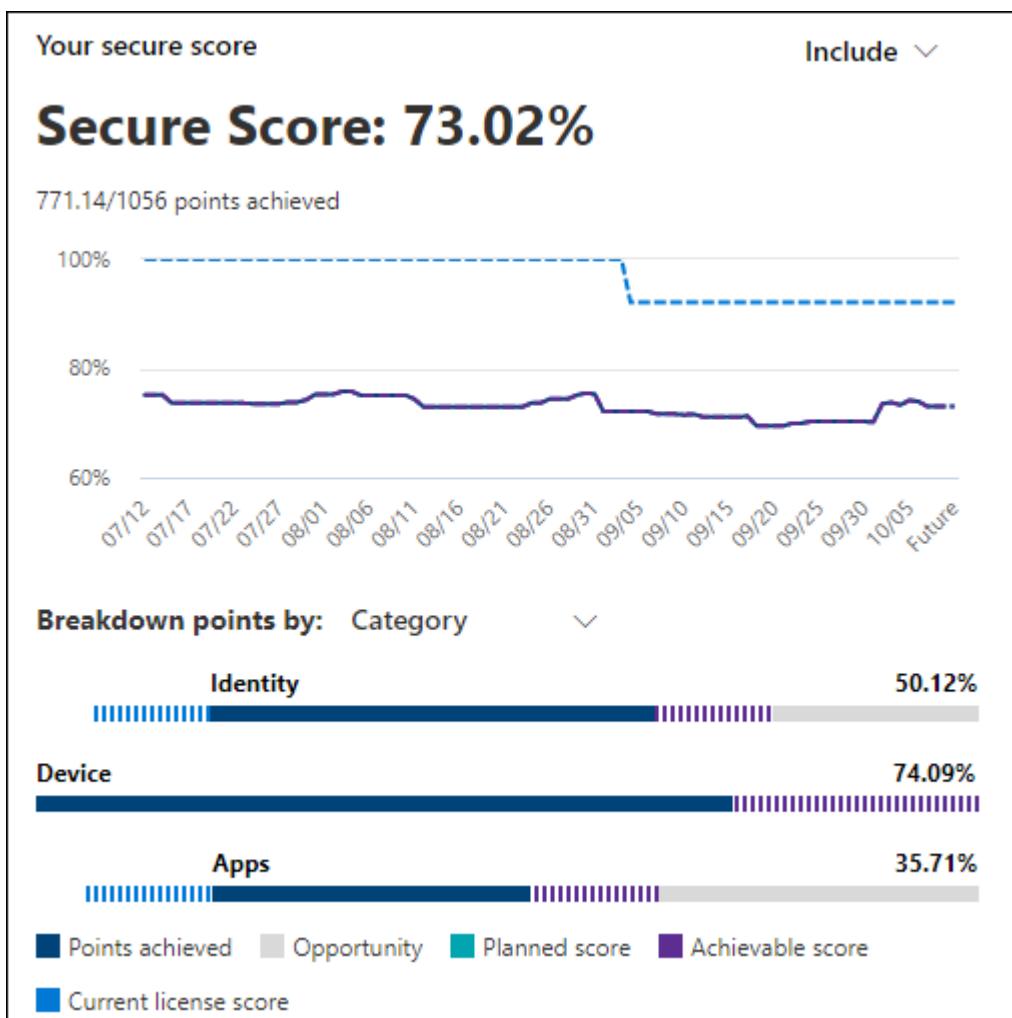
Since security defaults are enabled for newly created tenants by default, they will provide a good security baseline for new customers, which is good news since many customers are still not using any form of MFA and have the “old” default option (which is nothing at all) enabled.





2.3 Secure score

Implementing Conditional Access policies can help with receiving points in Secure Score (<https://securescore.microsoft.com/>). Secure Score provides a numerical summary of your security posture based on system configurations, user behavior and other security related measurements.



More information: [Microsoft Secure Score](https://securescore.microsoft.com/)

Conditional access provides functionality, which is part of a secure identity infrastructure, more actions than just implementing conditional access are needed. See this article for more information on the steps needed to secure your identity infrastructure: [Five steps to securing your identity infrastructure](#).





2.4 What's up with the preview label?

Sometimes you will see the label "Preview" added to certain Conditional Access functionality, which made me wonder if enabling these policies is supported in production environment. I reached out to Alex Simons, who is Corporate Vice President PM for Microsoft's Identity Division, providing the following answer. "Yes, all public preview features in Azure AD are fully supported".

 **Kenneth van Surksum**
@kennethvs

@alex_a_simons can you give me some guidelines on support for (Preview) options within #ConditionalAccess? IS there any support available when implementing these in production? #AzureAD

9:18 PM · Jul 23, 2019 · Twitter Web App

 View Tweet activity

 **Alex Simons** @Alex_A_Simons · 3m
Replies to @kennethvs
Yes, all public preview features in Azure AD are fully supported.

 1  1  1 

 **Kenneth van Surksum** @kennethvs · 31s
Great, thanks for the quick response!

Microsoft also has an official statement on everything that is under the preview status, and what you can expect from a support point of view. You can find that information here: [Supplemental Terms of Use for Microsoft Azure Previews](#)





2.5 Legacy authentication

When using Basic/Legacy Authentication application sends a username and password with every request to Exchange Online which either forwards the credentials towards Azure AD or a federated authentication provider like Active Directory Federation Services (ADFS). The problem with Basic/Legacy authentication is that it is vulnerable to brute force or password spray attacks. When legacy authentication is used, it is also possible to bypass Conditional Access.

Modern Authentication is based on OAuth 2.0 and the Active Directory Authentication Library (ADAL) providing token-based authentication. OAuth 2.0 in this case is the protocol being used, and ADAL is used to authenticate against Azure AD.

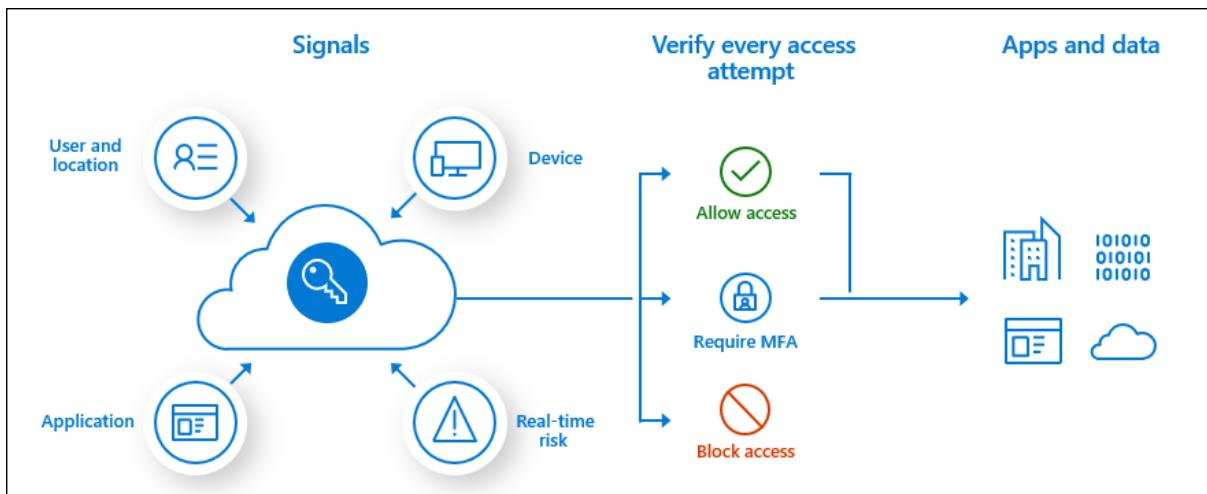
On September 23, 2021, the Exchange Team announced that **effective October 1st, 2022** basic authentication, regardless of usage will be permanently disabled in all tenants.

I've written an extensive article on how to that which you can read here: [Microsoft is going to disable basic/legacy authentication for Exchange Online. What does that actually mean and does that impact me?](#)





3 How does Conditional Access work?



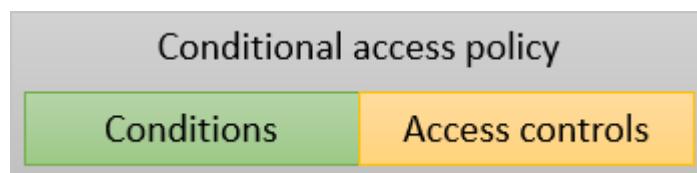
Microsoft [explains Conditional Access](#) in the following way.

Conditional Access consists of access scenario's called Conditional Access policies. An Conditional Access policy follows the following pattern:

When this happens	Then do this
-------------------	--------------

"**When this happens**" defines the reason for triggering your policy. This reason is characterized by a group of conditions that have been satisfied. With "**Then do this**" you define how users can access your cloud apps.

Technically this is translated to Conditions (When this happens) and Access controls (Then do this)



Microsoft provides some examples on their website for the most used Conditional Access policies, which you can use for reference. The following Conditional Access policies have been described:

1. [Block Legacy Authentication](#)
2. [Require MFA for administrators](#)
3. [Require MFA for Azure Management](#)
4. [Require MFA for all users](#)
5. [Sign-in risk-based Conditional Access](#)
6. [User risk-based Conditional Access](#)
7. [Secure security info registration](#)
8. [Block access by location](#)
9. [Require compliant devices](#)
10. [Block access](#)





3.1 Custom Conditional Access policies

Besides some of the common policies, customers can also create their own "custom" Conditional Access policies, the figure shows how a new Conditional Access policy are grouped into sections. The Conditions (When this happens) are grouped as assignments, and Access controls (Then do this) are grouped as Access controls.

The Microsoft description covers Conditional Access from a high-level overview, practically Conditional Access is a little more complex as explained in the following flowchart or cheat sheet. You can download this cheat sheet as PDF from the following location: [Conditional Access Workflow – v1.2.pdf](#)

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 *

Assignments

Users and groups [\(1\)](#)
0 users and groups selected

Cloud apps or actions [\(1\)](#)
No cloud apps or actions selected

Conditions [\(1\)](#)
0 conditions selected

Access controls

Grant [\(1\)](#)
0 controls selected

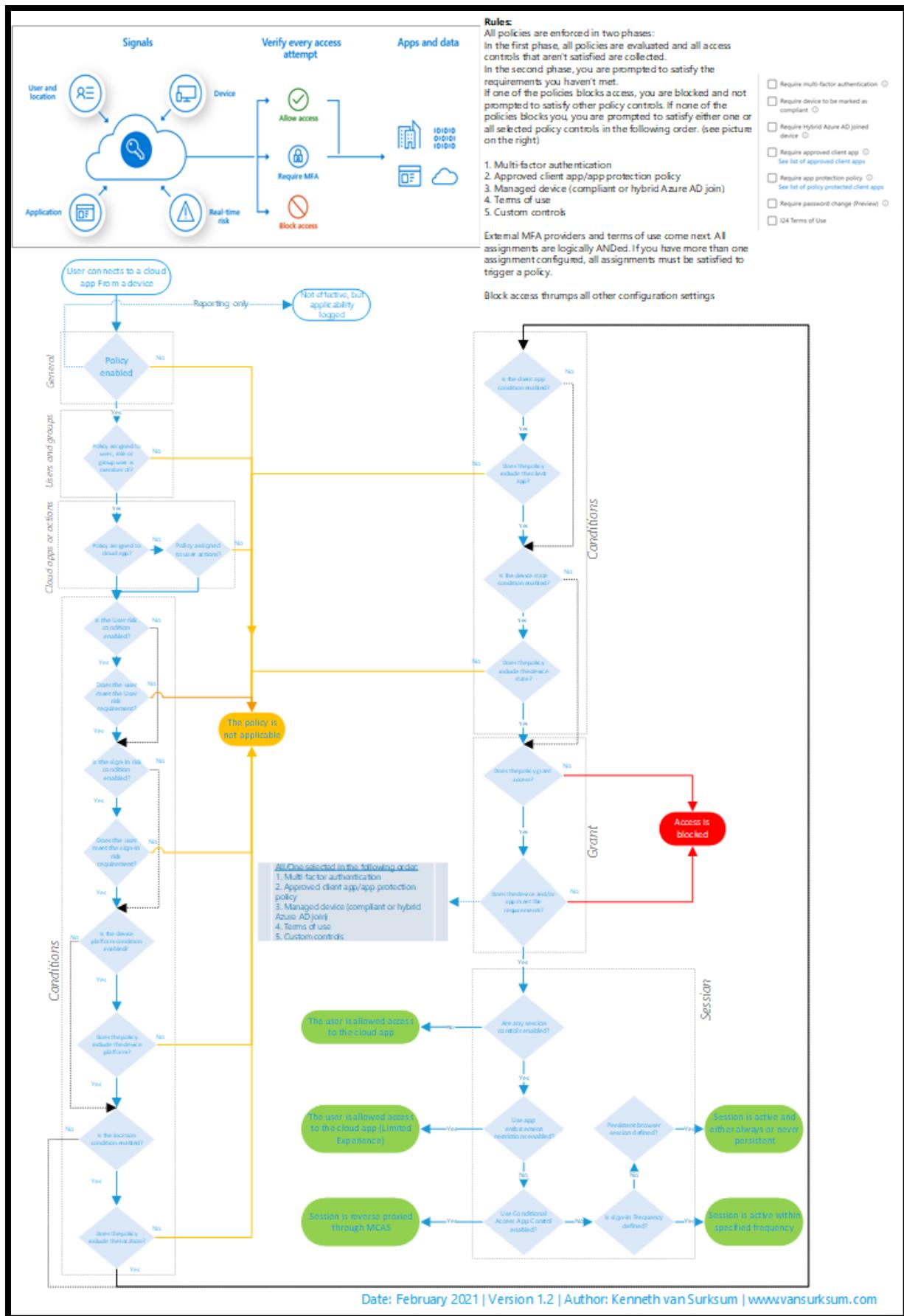
Session [\(1\)](#)
0 controls selected

Enable policy

Report-only On Off

[Create](#)







Based on the Conditional Access Workflow Cheat sheet, we can translate the conditional access to the following formula:

Conditional Access = Access to <provided> Cloud Apps except <provided> Cloud apps by <provided> users and/or <provided> roles and/or <provided> groups except <provided> users and/or <provided> groups using <provided> User Risk and/or <provided> Sign-in Risk and/or <provided> Device Platform except <provided> Device Platform from <provided> Location except <provided> Location using <provided> Client apps with <provided> device state, except <provided> device state Grants, Grants but <provided> requirement must be fulfilled> or Blocks access and/or applies Session controls.

When multiple Conditional Access policies apply for a user when accessing a cloud app, **all the policies must grant access** before the user can access the cloud app.

Some important rules are:

1. All policies are enforced in two phases:

- In the first phase, all policies are evaluated and all access controls that are not satisfied are collected.
- In the second phase, you are prompted to satisfy the requirements you have not met.
- If one of the policies blocks access, you are blocked and not prompted to satisfy other policy controls. If none of the policies blocks you, you are prompted to satisfy other policy controls, for which you require ALL or ONE of the selected controls, depending on the chosen option under “For multiple controls.” If ONE is selected, the following order is used while evaluating.
 - 1. Multi-factor authentication
 - 2. Approved client app/app protection policy
 - 3. Managed device (compliant or hybrid Azure AD join)
 - 4. Terms of use
 - 5. Custom controls

<input type="checkbox"/> Require multi-factor authentication (i)
<input type="checkbox"/> Require device to be marked as compliant (i)
<input type="checkbox"/> Require Hybrid Azure AD joined device (i)
<input type="checkbox"/> Require approved client app (i) See list of approved client apps
<input type="checkbox"/> Require app protection policy (i) See list of policy protected client apps
<input type="checkbox"/> Require password change (i)
<input type="checkbox"/> I24 Terms of Use

2. External MFA providers and terms of use come next. All assignments are logically ANDed. If you have more than one assignment configured, all assignments must be satisfied to trigger a policy.
3. Block access trumps all other configuration settings.

A Conditional Access policy is built from the following components:

- General
- Assignments





- Access controls

The different components are further described in the following paragraphs.

3.2 Structure of a Conditional Access policy explained

In this section we are going to explain how a Conditional Access policy is setup, there are some general settings like name on whether the policy is turned on, in report-only mode or off. And we have the Assignments and Access controls.

3.2.1 General

The conditional access policy must have a unique name, use a name which gives an idea of what the policy is doing under what circumstances.

Microsoft [recommends](#) the following naming structure for your Conditional Access policies

<SN>-	<Cloud app>:	<Response>	For	<Principal>	When	<Conditions>
-------	--------------	------------	-----	-------------	------	--------------

Within the name the following information should appear.

- A Sequence Number
- The cloud app(s) it applies to
- The response
- Who it applies to?
- When it applies (if applicable)

My personal recommendation is to further extend this suggested naming convention and device the policies into categories and version them.

For sequence numbers I use categories with a 3 digit follow up number:

- CAPxxx = Conditional Access Prerequisite
- CAUxxx = Conditional Access User
- CADxxx = Conditional Access Device
- CALxxx = Conditional Access Location
- CACxxx = Conditional Access Custom

And I also version the Conditional Access policy, allowing me to release new versions and test those on a few users first.

An example of my proposed naming convention would be:

CAD007-O365: Session set Sign-in Frequency for Apps for All users when Modern Auth Clients and Non-Compliant-v1.0





Policies can either be enabled (On), disabled (Off) or defined in Report-only mode which can help to determine whether the policy is working as supposed to. More on Report-only mode in chapter 5, Implementing Conditional Access.

3.2.2 Assignments

Assignments define the "When this happens" part of the Conditional Access rule and consists of the following conditions.

3.2.2.1 Users and Groups

In the users and groups condition you can specify for which users, Azure AD roles or groups the policy is applicable. You can either include and optionally exclude users, roles, and groups from the condition. It is also possible to include or exclude "guest and external users". Directory roles are especially interesting when using Azure Privileged Identity Management (PIM) where the Global Administrator role is assigned "temporarily" to a user instead of permanent. You can for example have a more restricting Conditional Access policy applied while the user has activated the Global Administrator rights.

Make sure that when defining your users that you always exclude your break glass accounts, and create a unique group for each conditional access policy, using the sequence number in its name so that you can allow an exception on your policy. Having this flexibility will help you, especially during initial rollout or migration to a new set of Conditional Access policies.

3.2.2.2 Cloud apps, user actions or authentication context

Microsoft defines a cloud app as a website, service or “endpoint protected by Azure AD Application Proxy”. The supported cloud apps from Microsoft can be found in the following list: [Microsoft cloud applications](#). Some of these cloud apps are Office 365, Office 365 Exchange Online, Office 365 SharePoint Online and Microsoft Azure Management (the Azure Portal). Besides the Microsoft applications you can also select any application which is configured as an Enterprise Application within your Azure AD.

Control user access based on all or specific cloud apps or actions. [Learn more](#)

Select what this policy applies to

Cloud apps

Include Exclude

None

All cloud apps

Select apps

The [Office 365 app](#) is a special application since it allows you to target all of the Office 365 services at once.

User actions refer to tasks a user can perform. For now, there are two user actions available:

- “Register security information”, which requires the user to register security information needed to start using MFA. More information on that here: [Combined security information registration for Azure Active Directory overview](#).
- Register or join devices, which is a newly introduced Conditional Access policy allowing to enforce Conditional Access during device join or registration (like asking for MFA). More Information here: [Using Conditional Access to provide more granularity when registering or joining devices](#)

Control user access based on all or specific cloud apps or actions. [Learn more](#)

Select what this policy applies to

User actions

Select the action this policy will apply to

Register security information

Register or join devices

Actions are a nice addition since they can help to make sure that prerequisites, like MFA are met, before enabling or applying other conditional access policies.





Keep in mind that if you do not use the User actions, you must select as least one cloud application for the Conditional access policy to work.

Authentication context, which at time of writing is still in public preview. Conditional Access Authentication Context can trigger a Conditional Access policy when sensitive content is accessed. With this new functionality new scenarios become available; some examples are:

- You can require MFA when a SharePoint site with a certain sensitivity label gets accessed on an unmanaged device.
- You can block the session from a Conditional Access policy when a SharePoint site with a certain sensitivity label gets accessed on an unmanaged device.
- You can require that Terms of Use must be accepted first before access to a SharePoint site with a certain sensitivity label gets accessed.
- You can set the sign-in frequency to a low value when a SharePoint site with a certain sensitivity label gets accessed.
- You can trigger the Authentication Context from within a session control policy defined in MCAS, for example to trigger MFA as defined in the Conditional Access policy. You could for example trigger MFA when a document with a certain sensitivity label gets downloaded or require the user to accept a Terms of Use first.
- App developers can leverage the authentication context in their own apps

Control user access based on all or specific cloud apps or actions. [Learn more](#)

Select what this policy applies to

Authentication context (preview) ▼

Authentication context is used to secure application data and actions in apps like SharePoint and Microsoft Cloud App Security. [Learn more](#)

Select the authentication contexts this policy will apply to

High Authentication Context

Medium Authentication Context

Low Authentication Context

More about Authentication Context later in this paper.

3.2.2.3 *Conditions*

There are many conditions you can use, and Microsoft is sometimes adding even more available conditions depending on new functionality becoming available in Azure AD. Below are the current conditions described.

3.2.2.3.1 *User Risk (additional license needed)*

If you have licensed Azure Active Directory Identity Protection as part of Azure AD Premium P2 you can use this condition as a criterion to determine to which situation the conditional access policy will apply. Azure Active Directory Identity Protection will generate a so called "User risk level" and based on the level (High, Medium, and Low) you can make the conditional access policy applicable. More information about this scenario here: [Conditional Access: Sign-in risk-based Conditional Access](#) and in an article I wrote about the subject here: "[Azure AD Identity Protection deep dive](#)"

User risk (Preview) ×

Configure ⓘ

Yes No

Configure user risk levels needed for policy to be enforced

High

Medium

Low

3.2.2.3.2 *Sign-in Risk (additional license needed)*





If you have licensed Azure Active Directory Identity Protection as part of Azure AD Premium P2 you can use this condition as a criteria to determine to which situation the conditional access policy will apply. Azure Active Directory Identity Protection will generate a so called "sign-in risk level" and based on the level (High, Medium, Low and No Risk) you can make the conditional access policy applicable. More information about this scenario here: [Conditional Access: Sign-in risk-based Conditional Access](#) and in an article I wrote about the subject here: "[Azure AD Identity Protection deep dive](#)"

Sign-in risk

Control user access to respond to specific sign-in risk levels. [Learn more](#)

Configure ⓘ

Yes No

Select the sign-in risk level this policy will apply to

High
 Medium
 Low
 No risk

3.2.2.3.3 [Device Platforms](#)

In the device platform condition, you can specify for which device platforms the policy is applicable. You can either include or optionally exclude device platforms from the condition. The following device platforms are available to select:

- Android
- iOS
- Windows Phone
- Windows
- macOS

You can also include All platforms, where you also include the platforms not in the list above (unsupported platforms, like for example Linux) and then exclude a certain supported platform from the list above. You can use the device platform condition in the case that you want to restrict access to cloud apps from managed devices, but also if you need to create several conditional access policies when you want to implement a feature which is not supported on all device platforms.

Device platforms

Apply policy to selected device platforms. [Learn more](#)

Configure ⓘ

Yes No

Include Exclude

Any device
 Select device platforms

Android
 iOS
 Windows Phone
 Windows
 macOS

Note: The device platform feature in Conditional Access is depending on user agent strings sent by the application or the web browser, which can easily be spoofed. This is something you must keep in mind when designing your Conditional Access policies. See this article from Nicola Suter for more excellent information: [Bypassing Conditional Access Device Platform Policies](#)

3.2.2.3.4 [Locations](#)

With locations you can specify conditions based on the network location the user is coming from, this is [always the public IP address](#) which is used on the internet and you cannot therefore use internal IP addresses to distinct in CA policies.

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 >

You can either include or exclude locations from the conditional access policy. Some use cases are that you want to restrict accessing the cloud app only from known locations (for example access to the Azure portal) or that you want to block access to a cloud app from a country or region for which you are sure your users will never use the cloud app service.

Note: Keep in mind that if within your company you provide guest network access but breakout to the internet using the same public IP address as your corporate devices, your guest network will fall under the same regime as your trusted network.





3.2.2.3.5 [Client apps](#)

Here you can specify the apps on the client for which the condition is applicable. These can be:

Modern Authentication Clients:

- **Browser** - apps accessed by a web browser on the client, see 4.2 for more information about restrictions.
- **Mobile Apps and desktop clients** - Applications supporting Modern Authentication

Legacy Authentication clients:

- **Exchange Active Sync clients** - apps which use Active Sync to connect to the cloud app - this option can only be selected if Exchange Online only is selected in the Cloud Apps selection. When Apply policy only to supported platforms is selected, only supported platforms like iOS, Android and Windows will be applicable.
- **Other clients** - apps which are not using "modern" authentication mechanisms, like IMAP, POP, SMTP etc... (for example, Outlook 2010). For more information about Modern and Legacy authentication see my article on that subject:

["Microsoft is going to disable basic/legacy authentication for Exchange Online. What does that actually mean and does that impact me?"](#)

Client apps

Control user access to target specific client applications not using modern authentication. [Learn more](#)

Configure ⓘ

Yes No

Select the client apps this policy will apply to

Modern authentication clients

Browser

Mobile apps and desktop clients

Legacy authentication clients

Exchange ActiveSync clients ⓘ

Other clients ⓘ

Done

3.2.2.3.6 [Device state](#)

When using the device state condition, you can exclude devices marked as compliant and devices which are Hybrid Azure AD joined (meaning Active Directory joined, and Azure AD registered) from the policy. Some scenarios are that you don't want the policy to apply to domain joined/azure ad registered devices, or that the managed device must report itself as compliant and if not, the policy will apply (block access for example until the device is compliant again)

Note: With Filter for devices now implemented, Microsoft recommends to move away from Device state and use Filter for devices instead.

Device state (Preview)

Control user access when device the user is signing-in from is not "Hybrid Azure AD joined" or "marked as compliant". [Learn more](#)

Configure ⓘ

Yes No

Include Exclude

Select the device state condition used to exclude devices from policy.

Device Hybrid Azure AD joined ⓘ

Device marked as compliant ⓘ

Done

3.2.2.3.7 [Filter for devices](#)

Filters for devices are available as conditions which you can use when creating your Conditional Access policies, with this functionality you can include or exclude devices based on filters using a rule expression. Combining include and exclude is not supported.

Microsoft provides the following [examples](#) where filters might provide a solution in their documentation:





1. Give access to Azure Management for privileged users only, coming from privileged or secure admin workstations
2. Block access from devices running non supported Windows versions (like Windows 7, 8.1)
3. Do not require MFA for specific account (traditional AD service accounts) when used on specific devices, like Teams phones or Surface Hub devices

The documentation states that Device state (which allows you to exclude Compliant and/or Azure AD Hybrid joined devices) **and** Filters for devices cannot be used in one Conditional Access policy. You can use the Compliancy and Azure AD Hybrid joined status in the Filter for devices as well though using the trustType and/or isCompliant properties, so basically this means that the Device State condition might disappear in the future to be replaced by the Filters for devices functionality.

Under the Filters for devices the following attributes can be used:

deviceId, displayName, manufacturer, mdmAppId, model, operatingSystem, operatingSystemVersion, physicalIds, profileType, systemLabels, trustType

Besides these attributes also custom attributes which can be set on the device in the form of extensionAttribute can be used, for this 15 custom fields can be used.

Each attribute supports operators, like Equals or Contains. But not every operator is supported for each attribute. The following operators are available: Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In and NotIn

The screenshot shows the 'Filters for devices (Preview)' dialog in the Azure Active Directory admin center. The left sidebar includes 'Dashboard', 'All services', 'FAVORITES' (with 'Azure Active Directory' selected), 'Users', and 'Enterprise applications'. The main area has a title 'Filters for devices (Preview)' and a sub-instruction 'Apply policy based on rule for device filters. [Learn more](#)'. A 'Configure' button has 'Yes' selected. Below it, 'Devices matching the rule:' has 'Include filtered devices in policy' selected. A table lists attributes: 'And/Or' (checkboxes for 'Add express' and 'Rule syntax'), 'Property' (dropdown showing 'Manufacturer' selected, with other options like 'DeviceId', 'DisplayName', etc.), 'Operator' (dropdown), and 'Value' (input field). The 'Manufacturer' option is highlighted in the dropdown. At the bottom are 'Done' and 'Edit' buttons.

3.2.2.3.7.1 Create include or exclude filter

The following table, which I copied from the documentation give a very good idea of the available options for each attribute. The examples give an example of what's possible for each attribute.

Supported device attributes	Supported operators	Supported values	Example





deviceId	Equals, NotEquals, In, NotIn	A valid deviceId that is a GUID	(device.deviceid -eq "498c4de7-1aee-4ded-8d5d-000000000000")
displayName	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	Any string	(device.displayName -contains "ABC")
manufacturer	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	Any string	(device.manufacturer -startsWith "Microsoft")
mdmAppId	Equals, NotEquals, In, NotIn	A valid MDM application ID	(device.mdmAppId -in ["0000000a-0000-0000-c000-000000000000"])
model	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	Any string	(device.model -notContains "Surface")
operatingSystem	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	A valid operating system (like Windows, iOS, or Android)	(device.operatingSystem -eq "Windows")
operatingSystemVersion	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	A valid operating system version (like 6.1 for Windows 7, 6.2 for Windows 8, or 10.0 for Windows 10)	(device.operatingSystemVersion -in ["10.0.18363", "10.0.19041", "10.0.19042"])
physicalIds	Contains, NotContains	As an example all Windows Autopilot devices store ZTDid (a unique value assigned to all imported Windows Autopilot devices) in device physicalIds property.	(device.devicePhysicalIDs -contains "[ZTDid]")





profileType	Equals, NotEquals	A valid profile type set for a device. Supported values are: RegisteredDevice (default), SecureVM (used for Windows VMs in Azure enabled with Azure AD sign in.), Printer (used for printers), Shared (used for shared devices), IoT (used for IoT devices)	(device.profileType -notIn ["Printer", "Shared", "IoT"])
systemLabels	Contains, NotContains	List of labels applied to the device by the system. Some of the supported values are: AzureResource (used for Windows VMs in Azure enabled with Azure AD sign in), M365Managed (used for devices managed using Microsoft Managed Desktop), MultiUser (used for shared devices)	(device.systemLabels -contains "M365Managed")
trustType	Equals, NotEquals	A valid registered state for devices. Supported values are: AzureAD (used for Azure AD joined devices), ServerAD (used for Hybrid Azure AD joined devices), Workplace (used for Azure AD registered devices)	(device.trustType -notIn 'ServerAD, Workplace')
extensionAttribute1-15	Equals, NotEquals, StartsWith, NotStartsWith, EndsWith, NotEndsWith, Contains, NotContains, In, NotIn	extensionAttributes1-15 are attributes that customers can use for device objects. Customers can update any of the extensionAttributes1 through 15 with custom values and use them in filters for devices condition in Conditional Access. Any string value can be used.	(device.extensionAttribute1 -eq 'SAW')

There are circumstances under which the Filter for devices conditions is applied or not applied depending on the Device registration state. So, it's important to realize that not everything that you build using the attributes and operators is supported. Microsoft describes these scenario in the following section of the documentation: [Policy behavior with filters for devices](#).

It states for example that for unregistered devices, when using positive operators like Equals, StartsWith, EndsWith, Contains or In for any attribute the device filter is not applied, but when using the negative operators like NotEquals, NotStartsWith, NotEndsWith, NotContains or NotIn for any attribute the device filter gets applied.

Which while reading it from the documentation doesn't make sense, let's see if it makes more sense if we define an example. Let's use the attribute manufacturer for that.

So based on the documentation the filter is not applied when we define that the device manufacturer should start with Microsoft, but the filter is applied when the device manufacturer property doesn't start with Microsoft.

It can also mean though that the filter gets applied for negative operators on unregistered devices because the attribute cannot be determined. This means for example that in the case that the unregistered device has the device manufacturer attribute of Microsoft and the operator is NotEquals Microsoft. the filter will apply because the attribute "device manufacturer" cannot be detected on the unregistered device. If this is the case, this is a caveat that must really be made clearer.





3.2.3 Access Controls

Access Controls define the "then do this" part of the conditional access policy. Based on the conditions the policy can:

- Block access to the cloud app
- Grant access to the cloud app
- Grant access to the cloud app but require an additional control (either one or all) from a list of selected controls (like MFA, must be compliant, Azure AD joined)

The grant access controls available are:

3.2.3.1 Grant

- Require multi-factor authentication (MFA)
 - Users are required to provide an extra authentication before access is granted
- Require device to be marked as compliant
 - Device from which the user is accessing the cloud app must be managed and compliant
- Require Hybrid Azure AD joined device
 - Device is Hybrid Azure AD joined, meaning member of Active Directory, and registered in Azure AD
- Require approved client app
 - Approved client apps are apps which can be managed using MAM functionality in Intune, for a list of supported apps see: [Supported mobile applications and desktop clients](#)
- Require app protection policy
 - Here you can specify that besides the fact that the application must be capable of being managed using MAM, that also app protection policies must have been applied.
- Optional: Terms of use, or other custom controls

<input type="checkbox"/> Require multi-factor authentication ⓘ
<input type="checkbox"/> Require device to be marked as compliant ⓘ
<input type="checkbox"/> Require Hybrid Azure AD joined device ⓘ
<input type="checkbox"/> Require approved client app ⓘ See list of approved client apps
<input type="checkbox"/> Require app protection policy (Preview) ⓘ See list of policy protected client apps
<input type="checkbox"/> I24 Terms of Use
For multiple controls
<input checked="" type="radio"/> Require all the selected controls
<input type="radio"/> Require one of the selected controls

Microsoft want to move towards App Protection policy checks. Therefore, we now see that some applications are supported under the Require App Protection Policy part, where these are not listed as Approved Client App. On the other hand, having the require Approved Client App functionality available could be handy if you only have Azure AD P1 and no Enterprise Mobility + Security license (giving you MAM capabilities).

The Microsoft documentation currently raises a lot of questions, when trying to figure out my options I decided to [create a matrix comparing the options](#) available:

- [Intune Protected Apps](#), the Official list that Microsoft references.
- [Approved Client App](#), the list of supported Apps when using a CA policy with Grant access controls
- [Require App Protection policy](#), the list of supported Apps when using a CA policy with Grant access controls.
- Applications listed under either iOS/iPadOS or Android when creating an App Protection Policy. (Keep in mind that some apps can be added using their custom id)

Based on this comparison, the only Apps with full support on all points are:





- Microsoft Edge
- Microsoft Excel
- Microsoft Lists (iOS)
- Microsoft Office
- Microsoft OneDrive
- Microsoft OneNote
- Microsoft Outlook
- Microsoft Planner
- Microsoft Power BI
- Microsoft PowerPoint
- Microsoft SharePoint
- Microsoft Teams
- Microsoft Word

3.2.3.2 Session Controls

Session controls are also part of the Access controls and can be applied after the session is granted, they allow for a limited experience within a cloud app and have the following options:

- [Use app enforced restrictions](#)

When this option is enabled, Conditional Access passes the device information to the cloud app, for now only SharePoint Online (SPO) and Exchange Online (EXO). In the cloud app a limited or full experience is offered depending on the device information.

- [Use Conditional Access App Control](#)

Routes the session through Microsoft Cloud App Security, which protects data by applying access and session controls acting as a reverse proxy. Some examples are: Prevent data exfiltration, protect on download, prevent upload of unlabeled files, and monitor user session for compliance. The options available here are: Monitor only (preview), Block downloads (preview) and Use custom policy.

- [Sign-in frequency](#)

With sign-in frequency you can specify the period before a user is asked to sign in again when attempting to access a resource. You can either choose Hours (between 1 and 23) or days (between 1 and 365)

- [Persistent browser session](#)

A persistent browser session allows users to remain signed in after closing and reopening their browser window. With this control, which can only be set when all cloud apps are selected you can choose between Always persistent or Never persistent. Never persistent requires the user to login again after the browser window is closed.

- [Disable resilience defaults \(Preview\)](#)

Resilience defaults are automatically enabled for all new and existing policies, and Microsoft highly recommends leaving the resilience defaults enabled to mitigate the impact of an outage. Admins may disable resilience defaults for individual Conditional Access policies.

Session

Control user access based on session controls to enable limited experiences within specific cloud applications. [Learn more](#)

Use app enforced restrictions ⓘ

This control only works with supported apps. Currently, Office 365, Exchange Online, and SharePoint Online are the only cloud apps that support app enforced restrictions. Click here to learn more.

Use Conditional Access App Control ⓘ

Sign-in frequency ⓘ

Persistent browser session ⓘ

Disable resilience defaults (Preview) ⓘ





3.2.3.3 App enforced restrictions

App enforced restrictions are only supported with Exchange Online and SharePoint Online. When enabled for SharePoint Online, users without a compliant device will see the following when accessing SharePoint via a web browser. See “[Control access from unmanaged devices](#)” for more information.

 Your organization doesn't allow you to download, print, or sync using this device. To use these actions, use a device that's joined to a domain or marked compliant by Intune. For help, contact your IT department. [More info.](#)

If a browser session to Exchange Online is configured to use App enforced restrictions, what can be done in Outlook Web Access can be restricted, for example offline mode and downloads can be restricted. The restrictions are defined in a so called OWA Mailbox Policy which can be set using PowerShell. See “[Conditional Access in Outlook on the web for Exchange Online](#)” for more information on how to configure the Mailbox policy.

For more information about how to use Conditional Access App Control, please read the following article on my blog: “[Extending Conditional Access to Microsoft Cloud App Security using Conditional Access App Control](#)”

3.2.3.4 Control Azure AD Conditional Access policy behavior during an Azure AD outage

In December last year, Microsoft [announced](#) that per April 1, 2021 they updated their service level agreement(SLA) for Azure AD user authentication from 99.9% to 99,99%. While this might seem like a small update in reality it makes a difference of 473 minutes (in a year with 365 days). With 99.9% the allowed downtime was 525 minutes and with 99.99% this is reduced to 52 minutes. This is still inconvenient though if you experience downtime for let's say 45 minutes on Monday morning while everyone is starting their work.

One of the techniques that Microsoft rolled out in order to support the new uptime, was the Azure AD Backup Authentication service, which Microsoft describes as the following:

Azure AD Backup Authentication service runs with decorrelated failure modes from the primary Azure AD system. This backup service transparently and automatically handles authentications for participating workloads as an additional layer of resilience on top of the multiple levels of redundancy in Azure AD. You can think of this as a backup generator or uninterrupted power supply (UPS) designed to provide additional fault tolerance while staying completely transparent and automatic to you.

So, what the Azure AD backup authentication service basically does is issuing tokens to applications for existing sessions if there is an outage of the primary authentication service. New sessions, or authentications by guest users are not supported. Which brings us to Azure AD Conditional Access, since access tokens are re-evaluated by Conditional Access policies before issued. But when the Backup authentication service is used, not all conditions can be evaluated in real time.

Session X

Control user access based on session controls to enable limited experiences within specific cloud applications. [Learn more](#)

Use app enforced restrictions ⓘ

ⓘ This control only works with supported apps. Currently, Office 365, Exchange Online, and SharePoint Online are the only cloud apps that support app enforced restrictions. [Click here to learn more.](#)

Use Conditional Access App Control ⓘ

Sign-in frequency ⓘ

Persistent browser session ⓘ

Disable resilience defaults (Preview) ⓘ

By introducing a new session control in Conditional Access, called “Disable resilience defaults” it's now possible to let the policy block authentications in the case that the backup authentication





service is active. By default, this option is turned off, which means that “in this case” when conditions cannot be evaluated in real time or evaluated with data collected at the beginning of the user’s session that the authentication will not be blocked.

The following conditions cannot be re-evaluated during an outage:

- Group Membership
- Role Membership
- Sign-in risk
- User risk
- Country location

That means that all other conditions, like MFA can be evaluated and those policies will not be impacted by this session. If you want to block based on Sign-in/User risk and don’t want to make concessions even during an outage, then this option is for you.

Microsoft recommends keeping the resilience defaults enabled, in order to mitigate the impact of an outage. But in case you have a high secure environment you might want to consider using this new session control option to control the behaviour during an outage.

Testing the options will be a challenge though. Which might be a reason to leave the options as-is, in my opinion.





4 Designing a Conditional Access strategy

When you are designing a Conditional Access strategy we first need to start with an inventory of the environment, in the most ideal situation you would design and implement conditional access in a green field scenario, but I for sure never had that luxury before so it is better to assume that the customer is already using cloud apps and wants to implement conditional access as a security measure.

The points to be inventoried are (but not limited to):

- What kind of devices does the customer use to access cloud apps?
 - Are the devices company owned, and fully managed?
 - Are the devices user owned, and non-managed?
- What kind of applications are currently used to access cloud apps?
- Is this a green field implementation, or are the cloud apps already in use without any conditional access policies in action?
- Does the customer use Intune and which scenarios are built into Intune?
 - Mobile Device Management
 - Mobile Application Management
- Is every user treated equally when it comes to access to the cloud apps, or can we distinct personas with different requirements when it comes to Conditional Access?
- Which licensing is the customer using? You need E5 functionality for administrators at least nowadays.
- How are licenses being assigned to users (groups, directly)
- Are there any service accounts used that interact with the cloud apps?
- Is Modern Authentication already enabled for Exchange Online and Skype for Business online?
- Is the company storing password hashes in Azure Active Directory?
- Are there cloud apps depending on each other?

When it comes to licensing while administrating Microsoft 365 services, please be aware that there are some things you need to be aware of, see also this article on my blog: "[License requirements for administering Microsoft 365 services](#)"

Microsoft has a document available helping in planning setting up Conditional Access, called the "Azure Active Directory Conditional Access Deployment Plan". The document in word format can be downloaded from the following location: <https://aka.ms/CADPDownload>. Microsoft also provides planning documentation online at: Plan a Conditional Access deployment - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/plan-conditional-access>

When designing a Conditional Access strategy in my experience it is important to really think on a high level on what you want to accomplish. It is very easy to start creating Conditional Access all kinds of individual Conditional Access policy and get lost from what you wanted to accomplish along the way.

Based on my experience the main goal of implementing Conditional Access is that you want to prevent access to your company data in situations where you do not have control over the data. That means that ideally cloud apps can only be accessed by:

1. Devices which are under company control and are compliant.
2. Applications which are under company control and are compliant.





3. Browser sessions on managed devices where data can be stored locally.
4. Browser sessions on non-managed devices where data can only be opened in the browser session and no data is left behind on the device.

All the other scenario's possible are either to fulfill requirements to successfully use Conditional Access or are additional security measures like always enforcing MFA when Azure AD administrators log in. It might also be that you need some "temporary" conditional access policies while migrating to the designed situation.

Below are some example scenario's which can influence the outcome of your design.

- Scenario 1: Allow devices managed by Intune access all the cloud apps using Apps and Desktop Clients and Modern Authentication Clients if compliant

Access to "All Cloud Apps" by "Users with EMS License" using "Any" device platform" coming from "any location" using "Mobile Apps and Desktop Clients" or "Modern authentication clients" is allowed, but device must be compliant.

- Scenario 2: Only allow Apps we can manage to access cloud apps when device is not managed.

Allow users with EMS License using devices not managed by Intune to access (portion of) cloud apps, using clients which we can manage using MAM policies (approved clients list)

- Scenario 3: Allow browser access to all the cloud apps from a trusted location

When users access the cloud apps from a trusted location, they can login without using any additional form of authentication.

- Scenario 4: Allow browser access to all the cloud apps from an untrusted location but use MFA and restrict the browser session (when possible)

When users access the cloud apps from a non-trusted location, they can login but have to use MFA and when possible, the browser session is restricted.

- Scenario 5: Block browser access to all the cloud apps from some geographic areas

Users cannot access cloud apps from regions where the company does not operate.

Once you know your scenario's trying to model the conditional access policy in a spreadsheet, by doing this you can determine if policies can be combined, or if more than one policy needs to be created to meet the requirements of the scenario. Keep in mind that the less is more.

4.1 Service dependencies

Many cloud apps have dependencies to other cloud apps, Microsoft Teams is a good example since it also provides access to SharePoint Online, and Planner for example. When this situation occurs, you must know how the application will behave, since policies may be applied either early-bound or late-bound. See the following article with more information about this: "[What are service dependencies in Azure Active Directory Conditional Access?](#)"

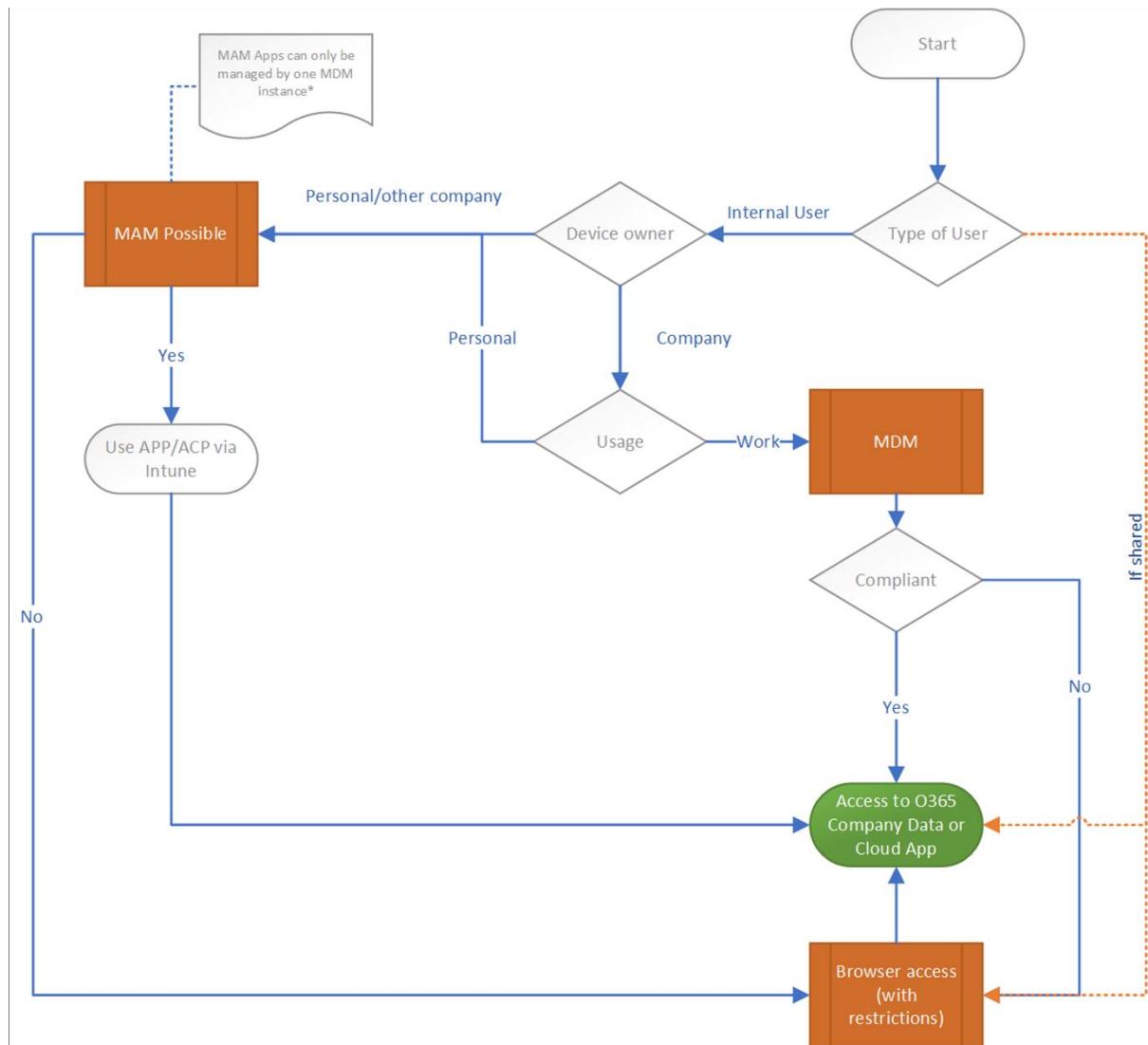




I've created a spreadsheet which can hopefully help you to document and write down your conditional access policies, the spreadsheet is available for download from the following location:
[Conditional Access Policy Description-v1.2.xlsx](#)

4.2 Functional Design

The functional flowchart below gives an overview of what I want to accomplish when designing an Conditional Access environment, I always use this flowchart and adjust it where needed to determine the use cases which must be supported.



As you can see, the flowchart is situated around giving access to company data, since I think that Data is the asset you must protect using Conditional Access policies.

So basically, we support several scenarios in this flowchart, let me describe some of them:

4.2.1 Guest users

In a default Microsoft 365 configuration, each user can invite Guest users into your Azure Active Directory. This is mostly done by either sharing a specific file hosted on OneDrive/SharePoint with





that Guest user, or by inviting that Guest user to a Microsoft Teams environment, where that Guest user can participate in the team.

With Conditional Access policies we can control how Guest users can access the environment. The options we have are:

4.2.2 Allow full access to the environment.

When you allow full access to the environment (which is the default), Guest users can use Desktop applications to access the data hosted by your company. In teams they can switch to your tenant and work in the teams that they are member of just like an internal user. They can even setup synchronization of SharePoint sites and have the data in that site available on their device.

You should ask yourself If you want to allow this, this totally depends on the data you are sharing with these guest users, in my opinion, if that data is confidential, you should not allow that data to be one a device which you not manage. It is also advised to implement a Governance procedure for Guest users and clean up occasionally, because without that Guest users can keep unlimited access the files shared and the Teams/SharePoint sites they have access to.

4.2.3 Allow Browser/Browser restricted access to the environment.

We can also disable access via Desktop clients and offer browser-based access only. In this way we have some better control since we can apply App Enforced Restrictions to the browser session and by doing so denying users the ability to download and print any company data. I've described this scenario in the following article: [Limit Access to Outlook Web Access, SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions](#). We can have the restrictions on all the data, or on just a part of the data by using the sensitivity label functionality for containers.

4.2.4 Device Owner and usage

When it comes to device owner and usage, we have several options:

4.2.4.1 *Company owned.*

Company owned devices in most cases are devices, mostly Windows and macOS laptops which are enrolled and managed using Microsoft Endpoint Manager/Intune. The device has several policies applied, and we can measure if the device is compliant to our security policies by using Compliance policies. Mobile devices can also be enrolled and managed, but with the current capabilities of what MAM can offer, and the way these devices are being used, I see that less often. Even though there are endless possibilities to manage mobile devices using MDM, these implementations are complex and must also be maintained.

4.2.4.2 *Personally owned.*

Personally owned devices, are owned by the user. In my opinion it does not make sense to start managing those devices, and most of the time these devices aren't even suitable to be company managed. If you ask the user bringing the device if he or she wants IT to manage the device, they will probably say NO. The Bring Your Own Device (BYOD) principle was a nice idea, but when it comes to managing a device and measure its integrity by using compliance policies, in the end you have to make the choice on whether you want to manage the device as a company yes or no. Managing a device comes at a cost and you should really ask yourself if the benefits outweigh the costs.





4.2.4.3 Company owned, personally used

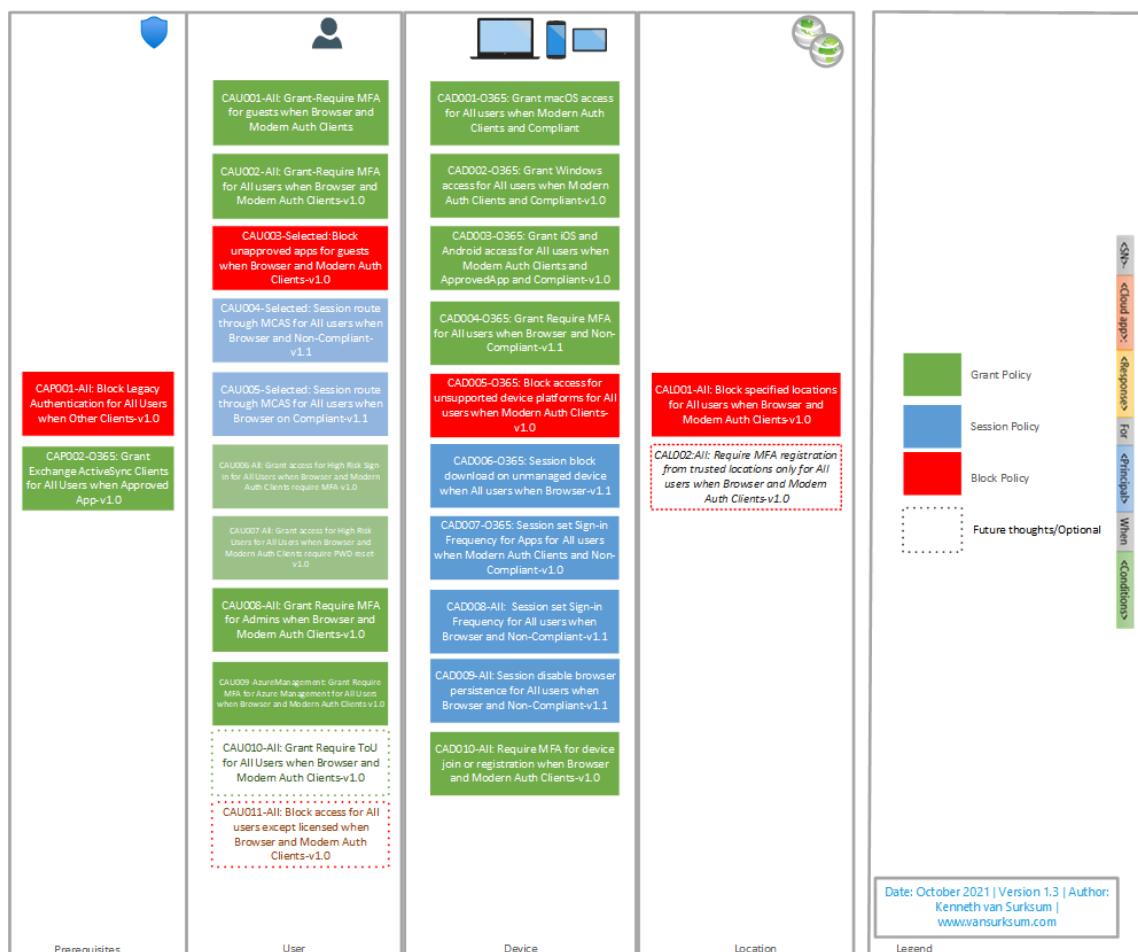
When it comes to mobile devices, this is the scenario encountered the most. Even though the company bought the device, the user using the device considers it personal. So besides hosting applications containing company data, the device will contain personal email, personal pictures, personal apps, and more personal data as well. If that is the status of the device, you will have a real hard story to tell your end users if you want to bring that device back under MDM and fully manage it from that point forward.

4.2.4.4 Other Company owned

If you work with external consultants but do supply those consultants with an Azure AD account (because you want those consultants to act on behalf of your company) it might be that those consultants already have a device managed by their own company. One of the limitations of MAM is that you cannot have more than one MDM solution managing the App. In that case the only option left over is to allow those external consultants to read their email using the web browser (which works quite well if you ask me).

The functional flowchart is my recommendation, it might be that you have another opinion or other requirements which require you to revise the flowchart. I think that the flowchart is a good starting point, in my work as a consultant I see many implementations where the IT department started with the Conditional Access policies, not having a clear idea on what they want to accomplish functionally.

Based on the Functional specifications I created a set of Default Conditional Access policies, these policies are described in more detail in chapter 5.





4.3 Browser restrictions and configuration when using Conditional Access.

Even though you are working in the browser on a compliant device, does not necessarily mean that Azure AD can detect that. Therefore, you must make sure that your browsers are configured correctly before you implement the Conditional Access policy.

Some examples I often encounter: End user is working on a compliant device, but cannot download or print files when using the web interface to connect to SharePoint online, this is caused by the App Enforced Restrictions policy being active (see: [Limit Access to Outlook Web Access, SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions](#)). Or, MCAS blocks the download of a file, even though the user is working on a compliant device. (see: [Extending Conditional Access to Microsoft Cloud App Security using Conditional Access App Control](#))

4.3.1 Browser support

This all has to do with browser support and configuration, below is an overview of the requirements and what is, and what is not supported. Currently Microsoft [supports the following browsers](#):

Os	Browsers
Windows 10	Microsoft Edge, Internet Explorer, Chrome, Firefox 91+
Windows 8 / 8.1	Internet Explorer, Chrome
Windows 7	Internet Explorer, Chrome
iOS	Microsoft Edge, Intune Managed Browser, Safari
Android	Microsoft Edge, Intune Managed Browser, Chrome
Windows Phone	Microsoft Edge, Internet Explorer
Windows Server 2019	Microsoft Edge, Internet Explorer, Chrome
Windows Server 2016	Internet Explorer
Windows Server 2012 R2	Internet Explorer
Windows Server 2008 R2	Internet Explorer
macOS	Microsoft Edge, Chrome, Safari

4.3.1.1 Sign-in Logging

When users are using a non-supported configuration, this might reflect as followed in the Azure AD sign-in logging. As you can see the Conditional Access policy requires a compliant device before access to the resource can be given. And in this case, our test user Ferry was working on a compliant device (you have to take my word for it).

Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only	Additional Details	Troubleshooting and support	Token issuer type	Token issuer name
Date	1/26/2021, 2:53:10 PM			User	Ferry Kuhlman			Azure AD	
Request ID	7380648e-6c51-411e-adcf-988fd1ef6300			Username	fkuhlman@emshelden.nl			Token issuer name	
Correlation ID	f57506ad-f329-4d74-a520-733a83407d6b			User ID	6da5bcc3-09fc-4364-a2b5-21272e8949cc			Latency	223ms
Authentication requirement	Multi-factor authentication			Alternate sign-in name	fkuhlman@emshelden.nl			User agent	
Status	Failure			Application	Microsoft Office 365 Portal				
Sign-in error code	53000			Application ID	00000006-0000-0ff1-ce00-000000000000				
Failure reason	Device is not in required device state: {state}. Conditional Access policy requires a compliant device, and the device is not compliant. The user must enroll their device with an approved MDM provider like Intune.			Resource	Windows Azure Active Directory				
				Resource ID	00000002-0000-0000-c000-000000000000				
				Resource tenant ID	74dd76e1-527c-4cd5-9bec-0a5d7c45990e				
				Client app	Browser				





4.3.2 Mozilla FireFox

Starting with Firefox version 91, Mozilla is [now supporting Single sign-on support \(SSO\) and device-based Conditional Access](#) as announced by Microsoft in the [What's new in Azure Active Directory for August 2021](#). The feature is still in Public Preview from a Microsoft point of view, and considered Advanced and experimental from a Mozilla point of view.

For now the option must be enabled, this can be done by opening Firefox, going to the menu and by selecting settings, from there you can go to Privacy & Security and go to the Logins and Passwords section. From there you can enable the feature by selecting the checkbox in front of "Allow Windows single sign-on for Microsoft, work and school accounts".

The screenshot shows the Mozilla Firefox Settings window with the 'Privacy & Security' tab selected. In the 'Logins and Passwords' section, there is a checkbox labeled 'Allow Windows single sign-on for Microsoft, work, and school accounts'. This checkbox is checked and highlighted with a red rectangular border. Below the checkbox, there is a link 'Manage accounts in your device settings'.

4.3.2.1 Configuring the settings from Microsoft Endpoint Manager

Since Mozilla is also providing ADMX files for configuring its browser, we can use this functionality to configure the "Allow Windows single sign-on for Microsoft, work and school accounts" setting using a custom Configuration Profile.

I will not go into too much detail on how you can use a custom ADMX file and leverage its settings in a Microsoft Endpoint Manager configuration profile. If you want to know more about that, I suggest that you read the following article from fellow-MVP Peter Klapwijk: [Manage new ADMX Backed Windows 10 policies with Microsoft Intune](#) and the Microsoft documentation "[Win32 and Desktop Bridge app ADMX policy Ingestion](#)"

You can download the latest ADMX file for FireFox from the [Mozilla Github page](#). Mozilla even has a Knowledge base article explaining how to ingest and set settings which you can find here: "[Managing](#)





[Firefox with Microsoft Endpoint Manager \(Intune\)](#)". The OMAURI [configuration string and possible values can be found under the WindowsSSO section of the Readme.md file](#).

Based on the information provided above we can create our custom configuration profile in Microsoft Endpoint Manager.

4.3.2.2 Create the custom Device Configuration Profile

Name: W10 - CP - Mozilla Firefox Configuration - v1.0 (or any other name you want to provide)

Profile type: Custom

OMA-URI settings:

Name: FireFox ADMX Ingestion

Description: FireFox ADMX Ingestion

OMA-URI: ./Device/Vendor/MSFT/Policy/ConfigOperations/ADMXInstall/Firefox/Policy/FirefoxAdmx

Value (String): Copy the content of the firefox.admx file into the Value field

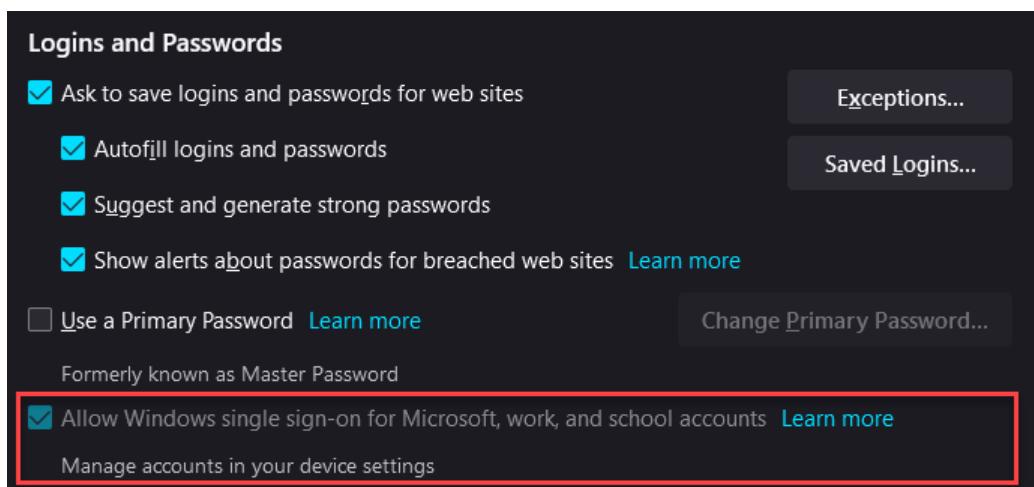
Name: Enable Windows SSO

Description: Enable Windows SSO

OMA-URI: ./Device/Vendor/MSFT/Policy/Config/Firefox~Policy~firefox/WindowsSSO

Value (String): <enabled/> <data id="WindowsSSO" value="1"/>

After the Configuration profile is successfully applied, you will notice that in FireFox the option is grayed-out meaning that it cannot be changed by the user.



4.3.3 Google Chrome

In order for the Google Chrome browser to support the device authentication you must deploy the [Windows 10 accounts extension](#) in the Chrome browser to your devices. You'll need this extension if you want to use the device compliancy within your Conditional Access policies.





The screenshot shows the Chrome Web Store interface. At the top left is the Google logo. To its right, the text "chrome web store" is displayed. Below this, a breadcrumb navigation path shows "Home > Extensions > Windows 10 Accounts". The main content area features a large blue Windows logo icon followed by the text "Windows 10 Accounts". Below the icon, it says "Offered by: Microsoft". A rating section shows "★★★★★ 390 | Productivity | 10,000,000+ users".

4.3.3.1 Deploying extensions for Google Chrome using Microsoft Endpoint Manager

You can configure the Google Chrome browser running on a Intune/MEM managed Windows 10 device by using a Configuration Profile with a custom profile type.

For this to work, we first need to [download the Google Chrome Bundle](#). Within that bundle you can find a folder called ADMX. From that folder you will need the chrome.admx file. Within the ADMX file we will use the ExtensionInstallForceList parameter to define the extensions we want to have installed.

```
<policy class="Both" displayName="$(string.ExtensionInstallForcelist)" explainText="$(string.ExtensionInstallForcelist_Explain)" key="Software\Policies\Google\Chrome" name="ExtensionInstallForcelist" presentation="$(presentation.ExtensionInstallForcelist)">
  <parentCategory ref="Extensions"/>
  <supportedOn ref="SUPPORTED_WIN7"/>
  <elements>
    <list id="ExtensionInstallForcelistDesc" key="Software\Policies\Google\ExtensionInstallForcelist" valuePrefix="/" />
  </elements>
</policy>
```

Secondly we must determine the unique identifiers for the extension(s) we want to install. You can determine this by browsing to the [chrome web store](#). From the chrome web store we need the Windows 10 Accounts extension, which at time of writing has the following id: "ppnbnpeolgkicgegkbkjmhlideopiji" make sure that if you are configuring this yourself to doublecheck whether the id still matches.





The screenshot shows the Chrome Web Store page for the 'Windows 10 Accounts' extension. The extension is offered by Microsoft and has over 10 million users. It has a rating of 3.97 stars from 397 reviews. A large yellow callout box highlights the feature: 'Instantly access accounts added to Windows without re-entering user credentials'. Below this, a 'Before' screenshot shows a 'Microsoft' sign-in button, and an 'After' screenshot shows a 'Microsoft' button labeled 'Sign in account'. At the top right of the page is a 'Sign in' link.

After downloading the ADMX file and figuring out which extensions we want to install by their ID in the chrome web store we can define our Configuration Policy.

First make sure that the custom policy ingests the ADMX file, use the following OMA-URI settings to configure this:

Name: Chrome ADMX Ingestion (can be any name, but make it easy to understand what it does)

Description: <Your description if you prefer>

OMA-URI:

./Device/Vendor/MSFT/Policy/ConfigOperations/ADMXInstall/Chrome/Policy/ChromeAdmx

Data type: String

Value: Copy/Paste here the contents of the ADMX file.

The 'Edit Row' dialog for OMA-URI Settings is shown. The 'Name' field contains 'Chrome ADMX Ingestion'. The 'Description' field contains 'Not configured'. The 'OMA-URI' field contains './Device/Vendor/MSFT/Policy/ConfigOperatio...'. The 'Data type' field is set to 'String'. The 'Value' field contains the following XML code:

```
<?xml version="1.0" ?>
<policyDefinitions revision="1.0"
schemaVersion="1.0">
<!--chrome version: 87.0.4280.141-->
<policyNamespaces>
<target
namespace="Google.Policies.Chrome" />

```

Secondly, we can define the setting we want to make as defined in the ingested ADMX file. Be very careful here.





Name: ExtensionInstallForcelist (can be any name, but make it easy to understand what it does)

Description: <Your description if you prefer>

OMA-URI:

./Device/Vendor/MSFT/Policy/Config/Chrome~Policy~googlechrome~Extensions/ExtensionInstallForceclist

Data type: String

Value: <enabled/> <data id="ExtensionInstallForcelistDesc"

value="1ppnbnpeolgkicgegkbkjmhlideopiji;https://clients2.google.com/service/update2/crx2bkbbeeffjeopflfhgeknacdiedcoml;https://clients2.google.com/service/update2/crx"/>

In the example above I also install another extension ([The Microsoft Defender Browser Protection](#)) as you can see the value must be carefully composed.

First, the ExtensionInstallForceList will eventually end up as a REG_MULTISZ registry string. Which means that each entry must be separated by the Unicode character 0xF000 (or  when encoded). You can also see that the URL <https://clients2.google.com/service/update2/crx> is being used. That URL is needed for the browser to determine the download path once its ready to download the extension. Each extension is also numbered, so the Windows 10 Accounts extension is number 1 and the Microsoft Defender Browser Protection extension is number 2. If you want to add more use 3, 4, etc...

Edit Row

OMA-URI Settings

Name *	ExtensionInstallForcelist
Description	Not configured
OMA-URI *	/Device/Vendor/MSFT/Policy/Config/Chrome...
Data type	String
Value *	<enabled/> <data id="ExtensionInstallForcelistDesc" value="1ppnbnpeolgkicgegkbkjmhlideopiji;https://clients2.google.com/service/update2/crx2bkbbeeffjeopflfhgeknacdiedcoml;https://clients2.google.com/service/update2/crx"/>

Once configured assign the policy to a group and you verify whether the necessary extensions are installed within the Google Chrome browser.

4.3.4 Microsoft Edge

Microsoft Edge obviously supports device authentication, but whether this is being used is depending on the profile you are signed into. When you are signed into a Microsoft Edge profile with enterprise Azure AD credentials, Microsoft Edge allows seamless access to enterprise cloud resources protected using Conditional Access. On a compliant device, the identity accessing the resource should match the identity on the profile. See: [Accessing Conditional Access protected resources in Microsoft Edge](#) for more information.





If you want to configure this sign-in for your devices you can use two settings using a Configuration Profile with an Administrative Template.

The first setting we must modify is the "Browser sign-in settings", make sure that the setting is enabled, and that the option "Force users to sign-in to use the browser" is selected.

Browser sign-in settings

Specify whether a user can sign into Microsoft Edge with their account and use account-related services like sync and single sign on. To control the availability of sync, use the 'SyncDisabled' (Disable synchronization of data using Microsoft sync services) policy instead.

If you set this policy to 'Disable browser sign-in', make sure that you also set the 'NonRemovableProfileEnabled' (Configure whether a user always has a default profile automatically signed in with their work or school account) policy to disabled because 'NonRemovableProfileEnabled' disables the creation of an automatically signed in browser profile. If both policies are set, Microsoft Edge will use the 'Disable browser sign-in' policy and behave as if 'NonRemovableProfileEnabled' is set to disabled.

If you set this policy to 'Enable browser sign-in' (1), users can sign into the browser. Signing into the browser doesn't mean that sync is turned on by default; the user must separately opt-in to use this feature.

If you set this policy to 'Force browser sign-in' (2) users must sign into a profile to use the browser. By default, this will allow the user to choose whether they want to sync to their account, unless sync is disabled by the domain admin or with the 'SyncDisabled' policy. The default value of 'BrowserGuestModeEnabled' (Enable guest mode) policy is set to false.

If you don't configure this policy users can decide if they want to enable the browser sign-in option and use it as they see fit.

* 0 = Disable browser sign-in
* 1 = Enable browser sign-in
* 2 = Force users to sign-in to use the browser

Setting type: Device

Supported on: Microsoft Windows 7 or later

Enabled Disabled Not configured

Browser sign-in settings

Force users to sign-in to use the browser

The second setting we must modify is called: "Configure whether a user always has a default profile automatically signed in with their work or school account". Make sure that this setting is enabled.

Configure whether a user always has a defa...

This policy determines if a user can remove the Microsoft Edge profile automatically signed in with a user's work or school account.

If you enable this policy, a non-removable profile will be created with the user's work or school account on Windows. This profile can't be signed out or removed.

If you disable or don't configure this policy, the profile automatically signed in with a user's work or school account on Windows can be signed out or removed by the user.

If you want to configure browser sign in, use the 'BrowserSignin' (Browser sign-in settings) policy.

Setting type: Device

Supported on: Microsoft Edge version 78, Windows 7 or later

Enabled Disabled Not configured

Finish the Configuration Profile and assign it to a group of your choosing.

On a sidenote, installing extensions in Microsoft Edge is much easier, since it is also part of the Administrative Templates, search for "Control which extensions are installed silently" and just supply the unique id or more in a list.





Control which extensions are installed silently

\Microsoft Edge\Extensions

Specifies extensions that are installed silently, without user interaction, and that the users can't uninstall or disable ("force-installed"). All permissions requested by the extensions are granted implicitly, without user interaction, including any additional permissions requested by future versions of the extension. Furthermore, permissions are granted for the enterprise.deviceAttributes and enterprise.platformKeys extension APIs. (These two APIs are only available to extensions that are force-installed.)

This policy takes precedence over a potentially conflicting 'ExtensionInstallBlocklist' (Control which extensions cannot be installed) policy. When you take an extension off of the forced-installed list it's automatically uninstalled by Microsoft Edge.

For Windows devices that aren't joined to a Microsoft Active Directory domain, forced installation is limited to extensions available in the Microsoft Store.

Note that users can modify the source code of any extension by using Developer Tools, potentially rendering the extension dysfunctional. If this is a concern, set the 'DeveloperToolsAvailability' (Control where developer tools can be used) policy.

Use the following format to add an extension to the list:

[extensionID]:[updateURL]

- extensionID - the 32-letter string found on edge://extensions when in developer mode.
- updateURL (optional) is the address of the Update Manifest XML document for the app or extension, as described at <https://go.microsoft.com/fwlink/?linkid=2095043>. If you don't set the updateURL, the Microsoft Store update URL is used (currently <https://edge.microsoft.com/extensionwebstorebase/v1/crx>). Note that the update URL set in this policy is only used for the initial installation; subsequent updates of the extension use the update URL indicated in the extension's manifest.

For example,
gggmmnlkjggpiggikcnhidnjhmcipibllhttps://edge.microsoft.com/extensionwebstorebase/v1/crx
installs the Microsoft Online app from the Microsoft Store "update" URL. For more information about hosting extensions, see: <https://go.microsoft.com/fwlink/?linkid=2095044>.

If you don't configure this policy, no extensions are installed automatically, and users can uninstall any extension in Microsoft Edge.

Note that this policy doesn't apply to InPrivate mode.

Example value:
gpbchcmnmhahfdphhkhkmpfmihenigjmpphttps://edge.microsoft.com/extensionwebstorebase/v1/cr
abködefghijklmnopabködefghijklmnop

Setting type: Device

Supported on: Microsoft Windows 7 or later

Enabled Disabled Not configured

Extension/App IDs and update URLs to be silently installed

bbcclinlkqijjkfjdpmiealijmmooekmp

4.4 Azure AD Conditional Access authentication context

During Microsoft Ignite in March this year, Microsoft announced several new upcoming functionalities for Azure Active Directory. One of the announcement was the Azure AD Conditional Access authentication context, for which I that time I already wrote about what it could do in my blog article: [Modern Workplace Management key takeaways from the Microsoft Ignite March 2021 announcements](#).

On May 26th, Conditional Access authentication was made available in public preview as announced by [Alex Simons](#) and [Caleb Baker](#) in the following article: [Conditional Access authentication context now in public preview](#).

4.4.1 Current data protection options

With Conditional Access, we are able to protect the identity of our Azure Active Directory users and control the access to company data hosted in Office 365 and SaaS applications. In order to protect the company data we have several options available.

We can allow full access on managed devices, these devices are either Azure AD Hybrid joined devices, or devices which are compliant and therefore managed by Microsoft Endpoint Manager. We can restrict access on non-managed devices, by either blocking or limiting the access given. We can protect the data itself, by using sensitivity labels which are either applied by the user, or automatically.





4.4.2 Conditional Access App Enforced Restrictions

We can use a Conditional Access policy, to either restrict or block access to data hosted in SharePoint Online on tenant or granular level. When the access control for unmanaged devices in SharePoint online is configured to Allow limited, web-only access you can restrict access to the content hosted in SharePoint online to be only available in a web browser where downloading and printing of an opened file is prohibited. This functionality can be accomplished by leveraging the Conditional Access App Enforced Restrictions option as part of the session access controls in a Conditional Access policy. I've described this functionality in the following article: [Limit Access to Outlook Web Access, SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions.](#)

If you require more granularity for your Conditional Access App Enforced Restrictions you can either specify the Control access from unmanaged devices setting on a per SharePoint site level using PowerShell or use Sensitivity labels, as described in the following article: [Defining more granularity for your Conditional Access App Enforced Restrictions using Sensitivity Labels](#)

4.4.3 Microsoft Cloud App Security (MCAS)

When our company data is not hosted on SharePoint online and Single Sign On (SSO) to a SaaS based 3rd party application is configured, we can route the users session to the third party application through Microsoft Cloud App Security (MCAS) which has a reverse proxy functionality (session control). Routing a user's session through MCAS is accomplished by leveraging the Conditional Access App Control functionality available under the session access controls.

Once a session is reversed proxied through MCAS we have several options available to control what can be done with files downloaded from the 3rd Party SaaS application. You can also decide not to use App Enforced Restrictions and use App Control instead for your SharePoint Online sites as well. I've explained some of the use cases in the following article before: [Extending Conditional Access to Microsoft Cloud App Security using Conditional Access App Control](#)

4.4.4 Conditional Access Authentication Context

Conditional Access Authentication Context is able to trigger a Conditional Access policy when sensitive content is accessed. With this new functionality new scenarios become available, some examples are:

- You can require MFA when a SharePoint site with a certain sensitivity label gets accessed on an unmanaged device.
- You can block the session from a Conditional Access policy when a SharePoint site with a certain sensitivity label gets accessed on a unmanaged device.
- You can require that Terms of Use must be accepted first before access to a SharePoint site with a certain sensitivity label gets accessed.
- You can set the sign-in frequency to a low value when a SharePoint site with a certain sensitivity label gets accessed.
- You can trigger the Authentication Context from within a session control policy defined in MCAS, for example to trigger MFA as defined in the Conditional Access policy. You could for example trigger MFA when a document with a certain sensitivity label gets downloaded or require the user to accept a Terms of Use first.
- App developers can leverage the authentication context in their own apps

So, how does it work?





4.4.5 Create Authentication context labels

The first thing that needs to be done in order to start working with Authentication context is to create a new label/new labels for authentication context. These labels can be created under the Authentication context (Preview) menu in the Conditional Access section of the Azure AD Admin portal. You can create a new label by clicking on + New authentication context which will open up a new windows where the name and description for the new label can be supplied. By default the Publish to apps option is selected, making the label available for use within apps.

To explain the concept I'm going to create the following authentication context labels

Label name	Explanation
Internal – High Authentication Context	Requires compliant device, TOU, or MFA
Internal – Medium Authentication Context	Requires compliant device
Internal – Low Authentication Context	MFA

With these labels we can build the following scenario's:

- Only grant access to internal users when they work on a compliant device, and require them to provide MFA
- Only grant access to internal users when they work on a compliant device
- Only grant access to internal users after they have provided MFA
- Only grant access to guest users after they accepted the Terms of Use (TOU), and require them to provide MFA
- Only grant access to guest users after they have provided MFA

Please be aware that at this time, you cannot delete any authentication context labels, you can modify existing ones though.





Creating the total solution requires the following steps:

- Create Authentication context label
- Apply authentication context applicability in sensitivity label, MCAS or App configuration
- Create Conditional Access policy which gets triggered if the authentication context applies.
-

4.4.6 Use the Authentication context label in your sensitivity labels

You can use authentication context in a sensitivity label. The authentication context in this case will replace the options which are available in SharePoint on how to handle unmanaged devices. So, you either need to choose between unmanaged device options in SharePoint (full access, limited web only access, or block access) or use authentication context.

When you create a new or modify an existing sensitivity label, you will get the screen similar to the one below.

The screenshot shows the 'Edit sensitivity label' interface in Microsoft 365 compliance. On the left, a vertical checklist indicates completed steps (Name & description, Scope, Files & emails, Groups & sites, External sharing & device access) and pending steps (Privacy & external user access, Azure Purview assets (preview), Finish). The main area, titled 'Define external sharing and device access settings', contains two main sections: 'Control external sharing from labeled SharePoint sites' and 'Use Azure AD Conditional Access to protect labeled SharePoint sites'. The 'Control external sharing' section offers four options: Anyone (selected), New and existing guests, Existing guests, and Only people in your organization. The 'Use Azure AD Conditional Access' section includes a checkbox and a dropdown menu for selecting an authentication context, with 'High Authentication Context - Requires compliant device, TOU, or MFA' selected. Navigation buttons at the bottom allow for backtracking, proceeding to the next step, or canceling.

The option to apply authentication context is only available within the Groups & sites section of the sensitivity label, which means that you can only define it on the container level and not on file level.

4.4.7 Creating the Conditional Access policy for the Authentication context label

So, now that the authentication context is added to the sensitivity label (sensitivity labels) we can define the Conditional Access policy which gets triggered at the moment that the SharePoint site which has the sensitivity label applied is accessed.

The Authentication context is a configurable action as part of the “Cloud apps or actions” section of the Assignments within a Conditional Access policy. Here you can see a list of the Authentication Context label which have been defined, notice that you can also use multiple authentication context labels at once.





CAU012-AuthCon: Require MFA for sensitivity label Confidential when nonCompliant-v1.0

...

X

Conditional Access policy



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

Control user access based on all or specific cloud apps or actions. [Learn more](#)

Select what this policy applies to

Authentication context (preview) ▾

Name *

Assignments

Users and groups ⓘ

All users included and specific users excluded

Cloud apps or actions ⓘ

1 authentication context included

Conditions ⓘ

1 condition selected

Access controls

Grant ⓘ

1 control selected

Session ⓘ

0 controls selected

Authentication context is used to secure application data and actions in apps like SharePoint and Microsoft Cloud App Security. [Learn more](#)

Select the authentication contexts this policy will apply to

High Authentication Context

Medium Authentication Context

Low Authentication Context

Not used - 1

Not used - 2

Not used - 3

Enable policy

Report-only On Off

Save

Below are some examples on possible Conditional Access policies which you can configure
In the example below we require both a compliant device **and** MFA for All users except Guest and External users (since we can't require them to have a compliant device)

CAU012-AuthCon: Require MFA and compliant for sensitivity label confidential-v1.0			Access Controls	
Assignments			Access Controls	
Users	Cloud Apps or Actions	Conditions	Grant	Session
All Users Except	Authentication Context: High Authentication Context	Client Apps: Browser Mobile Apps and Desktop Clients	Grant	
AAD_AA_ConAcc-Break-glass			Require multi-factor authentication And Require device to be marked as compliant	
AAD_AA_CAU012-Exclude				
All Guest and External Users				

With this policy active, from that point forward we can see that we can only access the SharePoint site from a Compliant device, with multi-factor authentication (1). We are not required to provide MFA, since the device I'm using for this test is using Windows Hello for Business (WHfB) which meets the MFA requirement.

If we would access the environment from a non-compliant device, we would receive the message as mentioned in (3), which reflects to the Failure described in (2). Keep in mind that also your browser must meet some requirements in order to be able to leverage this functionality, see: [Browser restrictions and configuration when using Conditional Access on your modern workplace](#).





1

2

3

For the conditional access policies we can create some variants, for example the policy below for Guest and External users which requires both MFA and acceptance of the Terms of Use.

CAU013-AuthCon: Require MFA and Accept TOU for sensitivity label confidential-v1.0		
Assignments		
Users	Cloud Apps or Actions	Conditions
All Guest and External Users Except	Authentication Context: High Authentication Context	Client Apps: Browser Mobile Apps and Desktop Clients
AAD_AA_ConAcc-Breakglass AAD_AA_CAU013-Exclude		

Access Controls	
Grant	Session
Grant	
Require multi-factor authentication And I24 - Terms of Use	

4.4.8 Impact on using App enforced restrictions

Since we can either choose between unmanaged device options in SharePoint (full access, limited web only access, or block access) or use authentication context. We cannot offer the option anymore to provide limited web only access and require MFA for example. Since we are using Authentication context in our Conditional Access policy, the App enforced restrictions option under session control in the Conditional Access policy is not available, since you can only use this option if you have Office 365, Exchange Online and SharePoint Online selected as a cloud app.

Use app enforced restrictions i

i This control only works with supported apps. Currently, Office 365, Exchange Online, and SharePoint Online are the only cloud apps that support app enforced restrictions. Click [here](#) to learn more.

So unless we reverse-proxy the session through Microsoft Cloud App Security, we lose the option to restrict users from downloading and printing files as part of App enforced restrictions. See: [Limit Access to Outlook Web Access, SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions](#) for more information about the functionality that





4.4.9 Use the Authentication context label in your MCAS Session Policy

We can also use authentication context in a Microsoft Cloud App Security session policy. A new action called “Require step-up authentication” has been added to the action list of a session policy.

Actions

Select an action to be applied when user activity matches the policy.

Test
Monitor login activities

Block
A default block message is displayed when possible

Protect
Apply sensitivity label to downloads & monitor all activities

Require step-up authentication PREVIEW ⓘ
Re-evaluate Azure AD Conditional Access policies based on the authentication context.
Unpublished authentication context will not be enforced
[Configure authentication context](#) ⏺

Always apply the selected action even if data cannot be scanned ⓘ

So, in the case of the scenario which I described before in the following article: [Extending Conditional Access to Microsoft Cloud App Security using Conditional Access App Control](#), we described the scenario where we reverse-proxied the SharePoint online connection through MCAS. Within MCAS we defined a session policy with a session control type of “Control file download (with inspection)” where we used the action “Block” if the Device was not Intune Compliant or Hybrid Azure AD joined, and the file name contains the word “BSN”

We can now extend this scenario to not block this file download, but trigger the Conditional Access policy which requires the approval of TOU and MFA.





4.5 Understanding and governing reauthentication settings in Azure Active Directory

Governing when users receive authentication prompts when authenticating to Azure Active Directory (Azure AD) is depending on more than one setting, on which functionalities are in use and in which scenario you authenticate (Browser, Modern clients or other). Reauthentication can take place by asking for a single factor, like password, FIDO, the [password less option in the Microsoft Authenticator app](#) or by using Multi Factor Authentication (MFA)

So you might understand that how reauthentication must be configured really depends per company and per scenario, so luckily Microsoft provides options which you can configure.

Some examples:

- You want users to reauthenticate more often when they come from a non-managed or non-registered device.
- You want users to reauthenticate more often when using a certain cloud application which you make available via Azure AD single sign on
- You might want some users in your organization to authenticate more often than others based on their risk profile.

4.5.1 The settings which make up the experience

Azure AD has a default sign-in frequency of 90 days, this might seem like a long time but there are some scenario's which require the user to sign in again, like:

- A change in the compliancy status of the managed device
- Disabling the account
- Revoking the sessions for the user
- Changing the password

Microsoft explains this default configuration as followed: "*don't ask users to provide their credentials if security posture of their sessions has not changed*", and states: "*If users are trained to enter their credentials without thinking, they can unintentionally supply them to a malicious credential prompt.*"

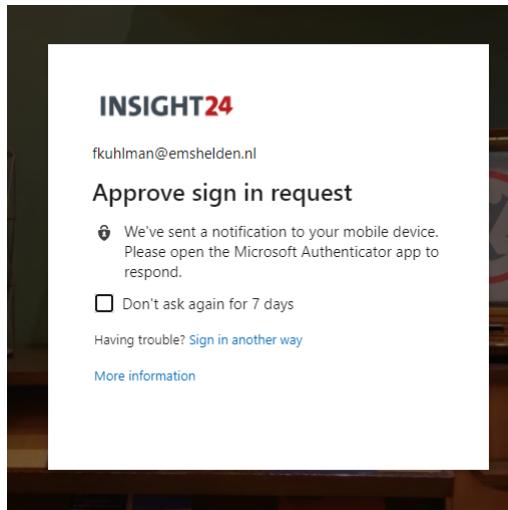
Personally, I think this default sign-in frequency makes sense when users work on managed devices, for other scenario's some adjustments might be necessary.

4.5.1.1 Azure Multi Factor Authentication Settings

When configuring Multi Factor Authentication you have the option to [remember the multi factor authentication](#) on trusted devices. When configured, the option allows you to bypass verifications for a specified number of days.

When using the Browser this is achieved by setting a cookie which expires after the specified time. In the screenshot below this has been set to 7 days.



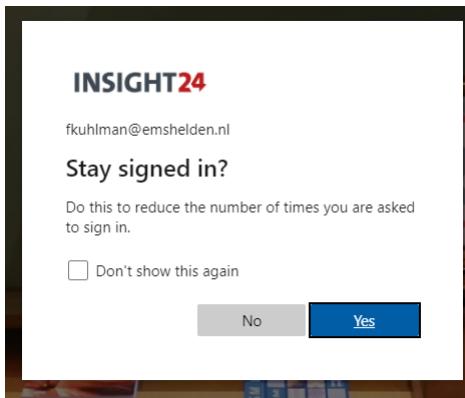


Non-browser apps use refresh tokens with a default validity of 1 hour, while validating the refresh token the check for MFA is performed as well.

Microsoft recommends that you set this setting to 90 days, in line with the default sign-in frequency. If needed, you must [revoke the MFA session](#) to force the user to re authenticate using MFA.

4.5.1.2 Show option to remain signed in (KMSI)

Another option influencing the experience is the option "Show option to remain signed in, also known as "Keep me signed in (KMSI)" for which the configurable settings can be found in a really strange place, the Company Branding settings.



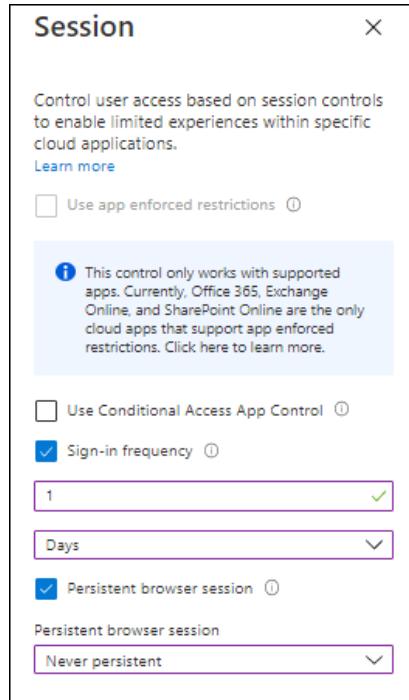
When users are presented with the Stay signed in option but abandon it, this is reflected in the Azure AD logging with a status of "Interrupted" and error code 50140.

Basic Info	Location	Device Info	Authentication Details	Conditional Access	Report-only	Additional Details
Date	10/22/2020, 12:28:15 PM			User	Ferry Kuhlman	Token issuer type: Azure AD
Request ID	28af3cb4-75bd-41aa-b65c-e60fe9e94700			Username	fkuhlman@emshelden.nl	Token issuer name
Correlation ID	198eaaf6-0976-45cb-8ad0-5cf9fc95b527			User ID	6da9bcc3-09fc-428d-a2b5-21272a8949cc	Latency: 216ms
Authentication requirement	Multi-factor authentication			Alternate sign-in name	fkuhlman@emshelden.nl	User agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.80 Safari/537.36 Edg/86.0.622.48
Status	Interrupted			Application	O365 Suite UX	
Sign-in error code	50140			Application ID	4345a709-9a63-4910-a426-33363201d503	
Failure reason	This occurred due to 'Keep me signed in' interrupt when the user was signing in.			Resource	Windows Azure Active Directory	
				Resource ID	00000002-0000-0000-0000-000000000000	
				Resource tenant ID	74dd7f61-527c-4cd5-9bec-0a5d7ca5990e	
				Client app	Browser	

4.5.1.3 Conditional Access

Conditional Access policies can be used to override some of the default settings in certain scenarios. By using session controls you can control how users must authenticate in different scenarios.





4.5.1.4 Sign-in Frequency

By setting the [Sign-in Frequency session control](#) you can override the default setting of 90 days to a lower setting, you can do this for example if users access your Office 365 environment from a non-managed device via the Browser, in the screenshot above we have set a sign-in frequency for 1 day.

See: [Policy 1: Sign-in frequency control](#) for an example on how to create a Conditional Access policy leveraging the sign-in frequency session control.

4.5.1.5 Persistent Browser session

A [persistent browser session](#) setting controls if users remain signed in after closing and reopening their browser window. We have 2 options here, either "Always persistent" or "Never persistent".

- This setting works correctly when "All cloud apps" are selected.
- This does not affect token lifetimes or the sign-in frequency setting.
- This will **override** the "Show option to stay signed in" policy in Company Branding.
- "Never persistent" will override any persistent SSO claims passed in from federated authentication services.
- "Never persistent" will prevent SSO on mobile devices across applications and between applications and the user's mobile browser.

See: [Policy 2: Persistent browser session](#) for an example on how to create a Conditional Access policy leveraging the "Persistent browser session" session control.

4.5.2 The scenarios which make up the experience

The scenarios under which users authenticate to your Azure AD environment are diverse, and you should understand which scenarios you will encounter and want to support within your organization. Below are some topics which should be considered when defining your scenarios.





4.5.2.1 *Supported applications.*

This sign-in frequency works with applications that have implemented Open Authorization (oAuth2) or OpenID Connect (OIDC) authentication protocols, which is supported for most applications working with Azure AD. Sign-in frequency also works with applications implementing the Security Assertion Markup Language (SAML) protocol for authorization and authentication as well.

When working in a Microsoft 365 modern environment you can assume that the Office desktop and mobile apps will work, also accessing the Office 365 web portals will support this without any issue. When it comes to 3rd party applications it depends. For example if an application drops its cookies for some reason and therefore redirects back to Azure AD, then the sign-in frequency can be less.

4.5.2.2 *Nonregistered devices*

If you work on devices which are not registered in Azure AD, it might also be that applications running on top of that device are not sharing their oAuth refresh token with each other, requiring the user to authenticate multiple times.

4.5.2.3 *Azure AD joined or Registered devices.*

Devices which are either Azure AD Joined, or Active Directory Joined/Azure AD registered via Hybrid AD join receive a so-called Primary Refresh Tokens (PRT) allowing them to use this token for Single Sign-on (SSO) functionality. A PRT is valid for 14 days and continuously renewed (every 4 hours at least) if the user actively uses the device. For more information about how the PRT works, I suggest to read the following article: "[What is a Primary Refresh Token?](#)" on the Microsoft website.

```
C:\WINDOWS\system32\cmd.exe
+-----+
| SSO State
+-----+
AzureAdPrt : YES
AzureAdPrtUpdateTime : 2020-10-22 05:04:02.000 UTC
AzureAdPrtExpiryTime : 2020-11-05 06:52:21.000 UTC
AzureAdPrtAuthority : https://login.microsoftonline.com/74dd76e1-527c-4cd5-9bec-0a5d7c45990e
EnterprisePrt : NO
EnterprisePrtAuthority :
```

4.5.3 Managed devices

Managed devices are devices on which you can measure compliance using Microsoft Endpoint Manager/Intune. Even though these devices are also registered, you also have the option to measure whether other security requirements are met, like for example BitLocker and Secure Boot being enabled on the device.





Windows 10 compliance policy

Windows 10 and later

✓ Basics 2 Compliance settings 3 Actions for noncompliance 4 Assignments 5 Review + create

Device Health

Windows Health Attestation Service evaluation rules

Require BitLocker Require Not configured

Require Secure Boot to be enabled on the device Require Not configured

Require code integrity Require Not configured

Device Properties

Configuration Manager Compliance

System Security

Microsoft Defender ATP

4.5.4 Browser used.

Which browser is used is an important factor when determining the scenarios. Is the browser being used an old browser like Internet Explorer, or a modern browser like Google Chrome, the new Microsoft Edge and Mozilla Firefox

4.5.5 Default settings when creating a new tenant.

When you create a new tenant today, the following default settings are available.

4.5.5.1 Multi factor authentication

Multi Factor Authentication by default is configured to not remember MFA on devices people trust. If you enable the setting the default number of days is set to 90 days.

remember multi-factor authentication on trusted device ([learn more](#))

Allow users to remember multi-factor authentication on devices they trust (between one to 365 days)

Number of days users can trust devices for

NOTE: For the optimal user experience, we recommend using Conditional Access sign-in frequency to extend session lifetimes on trusted devices, locations, or low-risk sessions as an alternative to 'Remember MFA on a trusted device' settings. If using 'Remember MFA on a trusted device,' be sure to extend the duration to 90 or more days. [Learn more about reauthentication prompts](#).

4.5.5.2 Show option to remain signed in

By default, in a new tenant, Company branding is not enabled and therefore the setting "Show option to remain signed in" is off.





The screenshot shows the 'Configure company branding' section of the Azure Active Directory settings. On the left, there's a sidebar with various Azure services like Devices, App registrations, Identity Governance, etc. The main area shows a status message 'Not configured' and instructions to 'Configure the text and graphics your users see when they sign in to Azure Active Directory.' There are fields for 'Banner logo', 'Square logo', 'Username hint', 'Sign-in page text', and 'Advanced settings' for 'Sign-in page background color' and 'Square logo image, dark theme'. At the bottom, there's an option to 'Show option to remain signed in' with 'Yes' and 'No' buttons.

4.5.5.3 Conditional Access

A new tenant does not have any Conditional Access policies configured. In the tenant I provisioned even the default security settings were not applied, but that can have something to do with the fact that I used a temporary tenant which was already hydrated.

The Default Security settings provide the following settings by default.

- Requiring all users to register for Azure Multi-Factor Authentication.
- Requiring administrators to perform multi-factor authentication.
- Blocking legacy authentication protocols.
- Requiring users to perform multi-factor authentication when necessary.
- Protecting privileged activities like access to the Azure portal.

For more information, see: [Microsoft deprecates Conditional Access baseline policies in favour of Security Defaults, here is what you need to know and do](#)

4.5.6 Bringing it all together

So now that we know the different options on how to configure reauthentication behavior and have an idea of the different scenarios we can face we can start designing our reauthentication scenarios.

4.5.6.1 Managed devices

For MDM managed devices, having the option to measure compliance gives us options to check whether the device is secure. Because of this I would advise to keep the defaults of the supplier in this case to keep the sign-in frequency to 90 days. MFA

4.5.6.2 Managed applications

For applications which you manage using Mobile Application Management (App Protection Policies) you can set a stricter sign-in frequency policy. For example, set this 7 days using a Conditional Access policy.





4.5.6.3 Non-managed devices

On non-managed devices (devices not compliant or not hybrid AD joined), especially when accessing the environment using the web browser (which is at this moment the real manageable option to keep your company data protection IMHO), you should even set a more restrict sign-in policy and disable browser persistence. An example would be to set the sign-in frequency to 1 day/4 hours and disable browser persistence.

Microsoft also provides some recommended settings depending on whether you have Azure AD Premium yes or no, which you can find here: [Recommended settings](#)





5 Implementing Conditional Access

Before you start implementing your Conditional Access policies you should define an implementation strategy, some things to consider are:

1. Make sure that Modern Authentication is enabled for Exchange Online (EXO) and Skype for Business Online (SfBO), SharePoint online has modern authentication enabled out of the box
2. Create 2 break glass accounts, these accounts, which are global administrator should have complex passwords and will be excluded from any conditional access policy created and must have MFA disabled (or at least on one of the two accounts). More information about creating break glass accounts can be found here: [Manage emergency access accounts in Azure AD](#).
3. For each conditional access policy created, we will create an exclusion group, so that we can deal with exceptions in our environment. These exception groups will be setup with Access review functionality (if available) to make sure that the memberships of these groups are evaluated on a regular basis.

Based on this we can define the following steps needed to implement your Conditional Access policies in the most ideal way.

Step 1: Check if modern authentication is enabled for Exchange Online and Skype for Business Online

Steps to check if modern authentication is enabled for Exchange Online and if not enable can be found here: [Enable modern authentication in Exchange Online](#)

Steps to check if modern authentication is enabled for Skype for Business Online and if not enable can be found here: [Enable modern authentication for Skype for Business Online](#) -

Step 2: Disable the legacy authentication protocols which are not used from the Office 365 accounts

This is not a small step since this can be a major change in your environment. Basically, it starts by monitoring how Exchange Online and Skype for Business Online are accessed, based on that determine the impact of this change. It might be that clients need to be updated to a newer version or that some special applications/services are accessing your environment using legacy protocols. The inventory should make this clear though and we have options to exclude certain accounts from our setup. In the end though it should be the goal to fully eliminate the use of legacy protocols.

Reference: [Protect Your Office 365 Accounts By Disabling Basic Authentication](#), and [Microsoft is going to disable basic/legacy authentication for Exchange Online. What does that actually mean and does that impact me?](#)

Step 3: Create 2 Break glass accounts.

If you do not have them already, create 2 break glass accounts which will be excluded from any policy which could potentially block access to the Azure environment in case something goes wrong

See the following article from Microsoft for more information: [Manage emergency access accounts in Azure AD](#)

Step 4: Implement your own custom Conditional Access policies based on your Conditional Access design, make sure that for every policy that you exclude a specific group for that policy and make





sure that the break glass accounts are excluded as well. Make sure that you have a decent test plan to test the policies and modify your Conditional Access policies if needed in case something does not work as expected.

Testing conditional access policies can be quite tricky. Main reason for this is that once a policy is enabled it is not necessarily directly effective and it can take a while for it to become effective for your clients. This has to do with the fact that clients are not logging in all the time. Therefore, make sure that before you enable your Conditional Access policy that you know where you can find the logging (more on that in the next chapter) to centrally determine if something is possibly wrong. Also make sure that all company procedures are followed for a change like this.

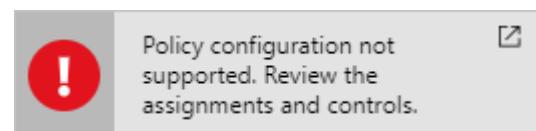
Step 5: Create a block all policy to make sure that you do not miss anything and that holes in your conditional access strategy exist. The way this works is that you create the policy but exclude all the groups used (include and exclude groups) for the Conditional Access policies and of course exclude the break glass accounts. In this case we are **absolutely sure** that all users are covered by Conditional Access policy

Users not authorized within any of the Conditional Access groups will not be able to sign in from that point forward.

The screenshot shows a computer monitor displaying a login page for 'INSIGHT24'. A red error message box is overlaid, stating: 'U hebt geen toegang tot deze dienst. U moet een andere gebruiker worden om deze dienst te kunnen gebruiken.' Below this, a smaller message says: 'Als u niet kunt inloggen omdat u een ander account heeft, neem dan contact op met de IT-dienst voor hulp.' To the right of the monitor is a detailed log entry in a table:

Basic info		Device info	MFA info	Conditional Access	Troubleshooting and support
Request ID	8c26caa9-b15a-4443-96c7-c76467590500				IP address [REDACTED]
Correlation ID	318dbf13-749c-4c98-a3d0-8e356be7d715				Location Langbroek, Utrecht, NL
User	Stanley Messie				Date 7/26/2019, 8:16:24 AM
Username	somesie@emsheldens.nl				Status Failure
User ID	f986377c-bdc3-46b4-804a-501150f581f8				Sign-in error code S3003
Application	Microsoft Office 365 Portal				Failure reason Access has been blocked due to conditional access policies.
Application ID	00000006-0000-0ff1-ce00-000000000000				Client app Browser
Resource	Windows Azure Active Directory				
Resource ID	00000002-0000-0000-c000-000000000000				

Azure has a safety feature that prevents you from creating a policy which violates the best practices for Conditional Access policies, so you cannot enable a policy for all cloud apps, for all users which denies access.



The safety feature is necessary because block all users and all cloud apps have the potential to block your entire organization from signing on to your tenant. You must exclude at least one user to satisfy the minimal best practice requirement.

Some other things to consider are:

For all the exception groups make sure that you enable the "Access review" functionality (Azure AD Premium P2 feature), for which you can find more information here:

- [Use Azure AD access reviews to manage users excluded from Conditional Access policies](#) -
- [Create an access review of groups or applications in Azure AD access reviews](#)
- [Which users must have licenses?](#)

Make sure that you have defined operational procedures on what to do if certain functionality provided by Microsoft is down. A good example of this is the fact that in the past the Microsoft Multi Factor Authentication service has been down for a significant time. Having an operational procedure





which allows you to make cloud apps available only from on premises when that happens without using MFA by disabling the "standard" policy and enabling a "temporary" policy might be a good idea.

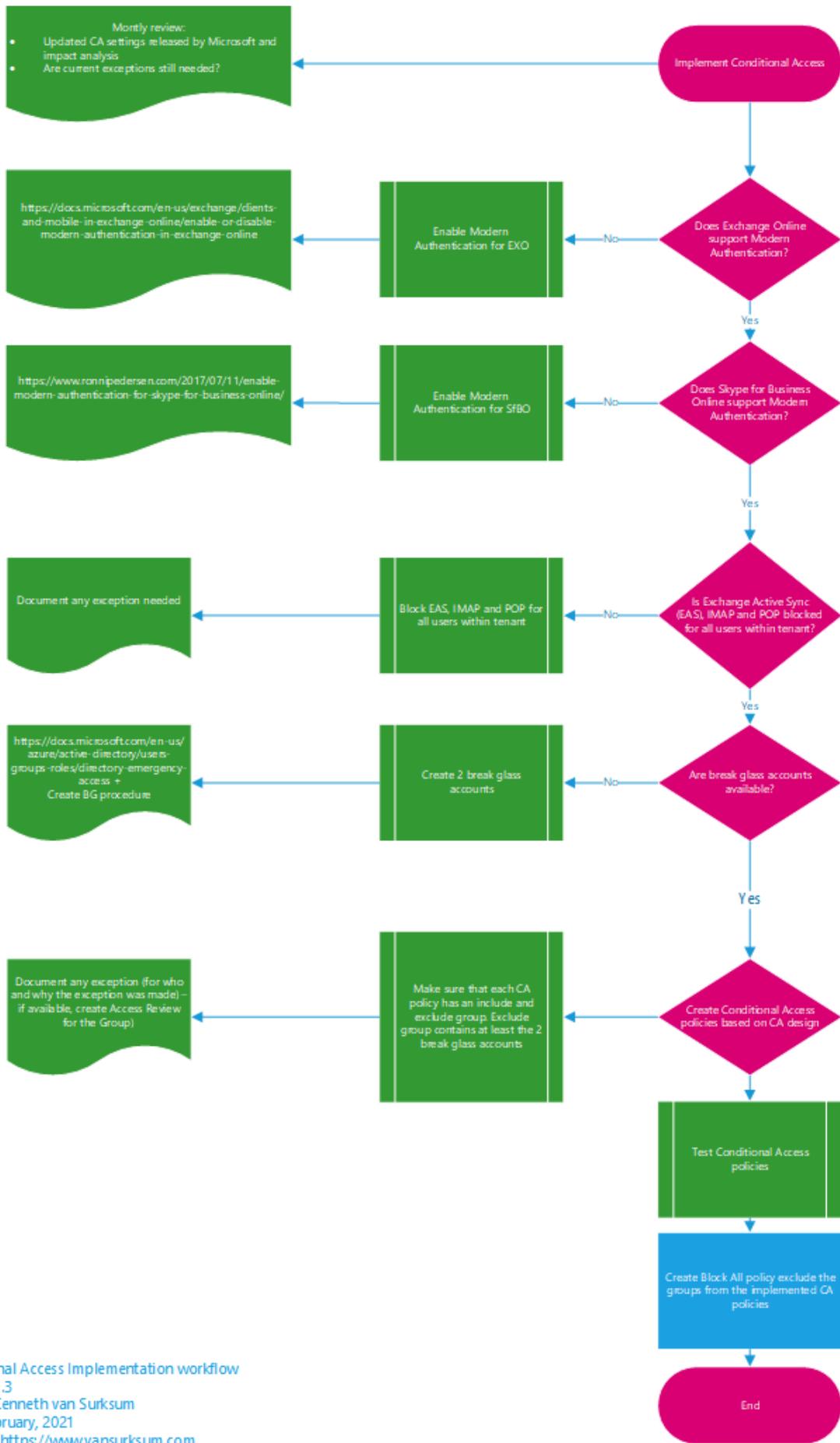
Azure Active Directory Conditional Access changes on a regular basis, make sure that you have a procedure to check occasionally what is cooking on this topic. Make sure you understand what is coming, what is in preview and what is released. Invest time to determine the possible impact of these changes to the Conditional Access policies and requirements in place. You can see [what's new in Azure Active Directory](#) and on the [Azure Updates webpage](#).

Some cloud apps have dependencies with other cloud apps, for example Microsoft Teams has dependencies of Exchange Online, SharePoint and Planner and perhaps even more.

More information here: [What are service dependencies in Azure Active Directory Conditional Access?](#)

I've created a flowchart which describe the steps described above, you can download this flowchart for your reference here: [Conditional Access Implementation Workflow - V1.3.pdf](#)





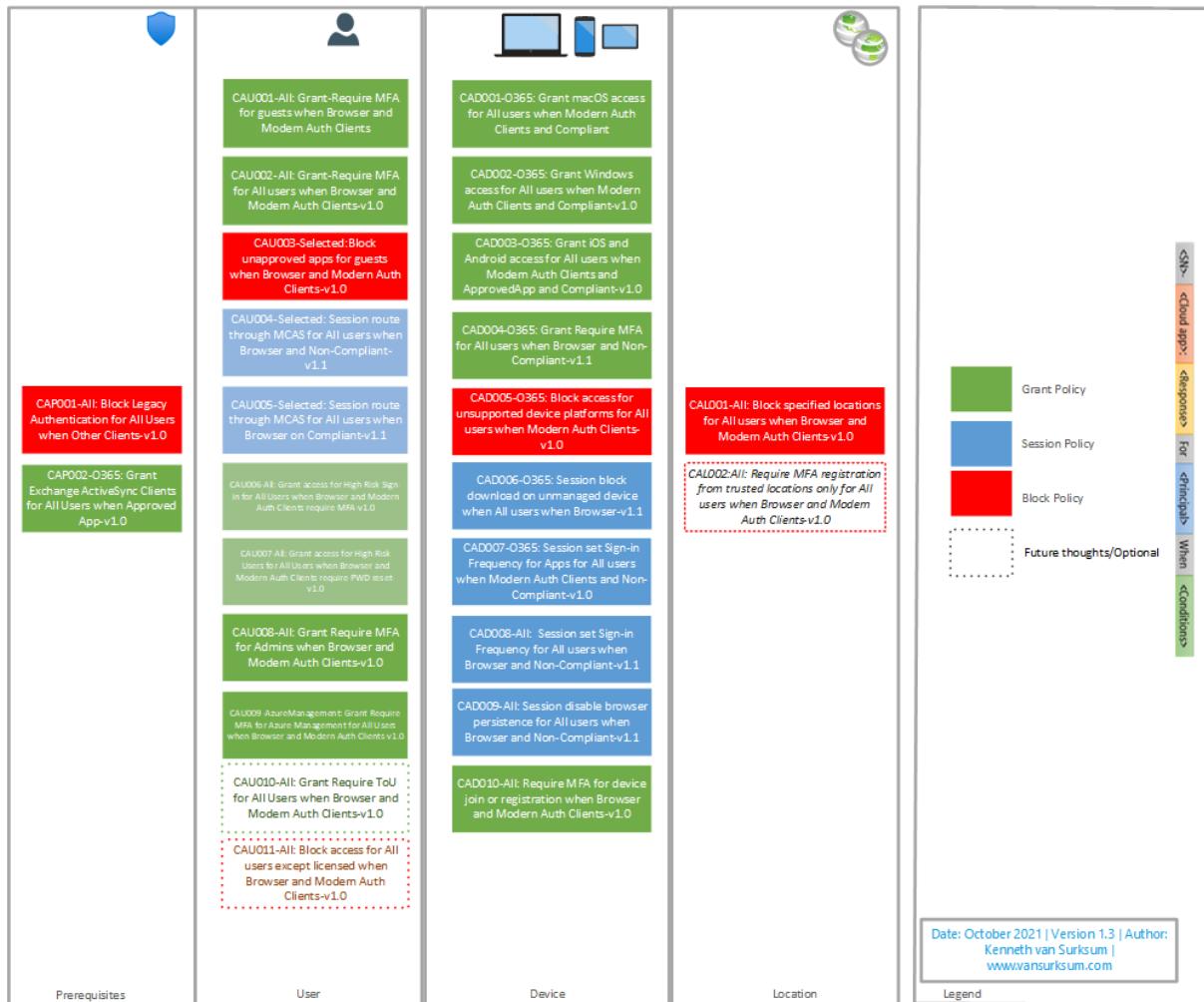
Conditional Access Implementation workflow
Version 1.3
Author: Kenneth van Surksum
Date: February, 2021
Website: <https://www.vansurksum.com>





5.1 My recommended default set of policies

Once you have consensus about how you want to allow access to your company data, you can start describing your Conditional Access policies, below is an overview of the Conditional Access policies based on the functional design described in chapter 4.



As you can see, the policies are divided into several categories which I use in the naming of the policies as well. For the naming I use the Microsoft recommended naming policy as described in this article: [Set naming standards for your policies](#)

<SN>- <Cloud app>: <Response> For <Principal> When <Conditions>

The detailed settings of the policies described, can be found in the new version of the Conditional Access Documentation spreadsheet which can be found here: [Conditional Access Policy Description-v1.2.xlsx](#)

For each policy, an exclusion group is created, and for each policy the group containing the break glass accounts will also be excluded from the policy.





5.1.1 Versioning

Each policy has a version number in it, by doing so we can update policies with a new version, test this new policy on a select set of users and/or Report-Only mode and then make the switch by turning the old version off, and implementing the new one.

5.1.2 Prerequisites

The first category are the prerequisites and contains two policies.

5.1.2.1 CAP001-All: Block Legacy Authentication for All users when OtherClients-v1.0

This policy blocks all Legacy authentication when clients not supporting Modern Authentication are being used. For more information about this policy, please read the documentation from Microsoft: [How to: Block legacy authentication to Azure AD with Conditional Access](#)

Keep in mind that just disabling Legacy authentication in an existing environment isn't a good idea. The end-users might still use applications only capable of performing legacy authentication and they might need your help first to transition to apps which support modern authentication. Please read the following article for more context and how to start your own project to phase out legacy authentication. [Microsoft is going to disable basic/legacy authentication for Exchange Online. What does that actually mean and does that impact me?](#)

CAP001-All: Block Legacy Authentication for All users when OtherClients-v1.0		
Assignments		Access Controls
Users	Cloud Apps	Conditions
All Users	All	Client Apps: Other
Except		
AAD_AA_ConAcc-Breakglass		
AAD_AA_CAP001-Exclude		

5.1.2.2 CAP002-O365: Grant Exchange ActiveSync Clients for All users when Approved App-v1.0

Based on the functionalities provided, there is no use case to keep using Exchange Active Sync, that is the reason that we block Exchange Active Sync clients as well. Even though the policy grants, it blocks access to EAS clients because an Approved App is needed. (Outlook in our case). By using this mechanism, users still using EAS receive a message that they must transition to an application supporting Modern Authentication. If we would use a block policy, the user simply will not get any email messages anymore.

CAP002-O365: Grant Exchange ActiveSync Clients for All users when Approved App-v1.0		
Assignments		Access Controls
Users	Cloud Apps	Conditions
All Users	All	Client Apps: Exchange Active
Except		
AAD_AA_ConAcc-Breakglass		
AAD_AA_CAP002-Exclude		





5.1.3 User

The second category contains the policies for users and contains nine policies. Some of these policies require an Azure AD Premium P2 subscription, like CAU005 and CAU006 which require Azure AD Identity Protection and CAU004 which requires Microsoft Cloud App Security (MCAS).

5.1.3.1 CAU001-All: Grant Require MFA for guests when Browser and Modern Auth Clients-v1.0

This one makes sure that guest users are required to use Multi Factor Authentication (MFA) when accessing resources that you host.

CAU001-All: Grant Require MFA for guests when Browser and Modern Auth Clients-v1.0		
Assignments		
Users	Cloud Apps	Conditions
Guest and External Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU001-Exclude	All	Client Apps: Browser Mobile Apps and Desktop Clients
Access Controls		
	Grant	Session
	Grant	
	Require multi-factor authentication	

5.1.3.2 CAU002-All: Grant Require MFA for All users when Browser and Modern Auth Clients-v1.0

This policy requires each user to use MFA when accessing cloud apps.

CAU002-All: Grant Require MFA for All users when Browser and Modern Auth Clients-v1.0		
Assignments		
Users	Cloud Apps	Conditions
All Users Except Guest and External Users And AAD_AA_ConAcc-Breakglass AAD_AA_CAU002-Exclude	All	Client Apps: Browser Mobile Apps and Desktop Clients
Access Controls		
	Grant	Session
	Grant	
	Require multi-factor authentication	

5.1.3.3 CAU003-Selected: Block unapproved apps for guests when Browser and Modern Auth Clients-v1.0

With this policy, you can create a list of Cloud Apps for which guest users are not allowed to use them. These apps can be apps containing sensitive company data for example.

CAU003-Selected: Block unapproved apps for guests when Browser and Modern Auth Clients-v1.0		
Assignments		
Users	Cloud Apps	Conditions
Guest and External Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU003-Exclude	<Selected>	Client Apps: Browser Mobile Apps and Desktop Clients
Access Controls		
	Grant	Session
	Block	

5.1.3.4 CAU004-Selected: Session route through MCAS for All users when Browser on Non-Compliant-v1.1

With this policy you can route the session through MCAS and its reverse-proxy capability, allowing you to either block downloads or monitor the session for strange behavior. In this example we block downloads for Cloud Applications specified if the device is not compliant.





CAU004-Selected: Session route through MCAS for All users when Browser on Non-Compliant-v1.1

Users	Assignments		Access Controls	
	Cloud Apps, User Actions or Authentication Context	Conditions	Grant	Session
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU004-Exclude	<Selected>	Client Apps: Browser Filter for devices: Exclude isCompliant Equals True		Use Conditional Access App Control: Block downloads (Preview)

5.1.3.5 CAU005-Selected: Session route through MCAS for All users when Browser on Compliant-v1.0

With this policy you can route the session through MCAS and its reverse-proxy capability, allowing you to either block downloads or monitor the session for strange behavior. In this example we just monitor the session on devices which are compliant.

CAU005-Selected: Session route through MCAS for All users when Browser on Compliant-v1.1

Users	Assignments		Access Controls	
	Cloud Apps, User Actions or Authentication Context	Conditions	Grant	Session
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU005-Exclude	<Selected>	Client Apps: Browser Filter for devices: Include isCompliant Equals True		Use Conditional Access App Control: Monitor (Preview)

5.1.3.6 CAU006-All: Grant access for High Risk Sign-in for All Users when Browser and Modern Auth Clients require MFA-v1.0

This policy will require MFA for sign-ins flagged as high-risk by Azure AD identity protection. See my article: [Azure AD Identity Protection deep dive](#) for more information.

CAU006-All: Grant access for High Risk Sign-in for All Users when Browser and Modern Auth Clients require MFA-v1.0

Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU006-Exclude	All	Client Apps: Browser Mobile Apps and Desktop Clients Sign-in Risk: High	Grant	
				Require multi-factor authentication

5.1.3.7 CAU007-All: Grant access for High-Risk Users for All Users when Browser and Modern Auth Clients require PWD reset-v1.0

This policy will grant access for High-Risk users after but only after they have reset their password. This functionality is also provided by Azure AD Identity Protection.

CAU007-All: Grant access for High Risk Users for All Users when Browser and Modern Auth Clients require PWD reset-v1.0

Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU007-Exclude	All	Client Apps: Browser Mobile Apps and Desktop Clients User Risk: High	Grant	
				Require Password Change

5.1.3.8 CAU008-All: Grant Require MFA for Admins when Browser and Modern Auth Clients-v1.0

This policy makes sure that Admins must always use MFA before signing into any cloud application.





CAU008-All: Grant Require MFA for Admins when Browser and Modern Auth Clients-v1.0

Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
User Account Administrator SharePoint Service Administrator Security Administrator Password Administrator Helpdesk Administrator Company Administrator Exchange Service Administrator Conditional Access Administrator Billing Administrator Authentication Administrator Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU008-Exclude	All	Client Apps: Browser Mobile Apps and Desktop Clients	Grant Require multi-factor authentication	

5.1.3.9 CAU009-AzureManagement: Grant Require MFA for Azure Management for All Users when Browser and Modern Auth Clients-v1.0

This policy makes sure that MFA is required when the Azure Management portal is requested. The reason for this policy is that when using PIM, your admin users might first go to the Admin portal, to request their rights using PIM afterwards. See: [Lessons learned while implementing Azure AD Privileged Identity Management \(PIM\)](#) for more information.

CAU009-AzureManagement: Grant Require MFA for Azure Management for All Users when Browser and Modern Auth Clients-v1.0

Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
All	Windows Azure Service Management API	Client Apps: Browser Mobile Apps and Desktop Clients	Grant	
Except				
AAD_AA_ConAcc-Breakglass AAD_AA_CAU009-Exclude			Require multi-factor authentication	

5.1.3.10 CAU010-All: Grant Require ToU for All Users when Browser and Modern Auth Clients-v1.0 (Optional)

This one is optional, but the policy requires users to agree to the Terms of Use (TOU) first, before they are allowed to access the resources.

CAU010-All: Grant Require ToU for All Users when Browser and Modern Auth Clients

Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
All Users	All	Client Apps: Browser Mobile Apps and Desktop Clients	Grant	
Except				
AAD_AA_ConAcc-Breakglass AAD_AA_CAU010-Exclude			Terms of Use	

5.1.3.11 CAU011-All: Block access for All users except licensed when Browser and Modern Auth Clients-v1.0 (Optional)

This one is optional as well, but I personally recommend it even though it is a risky one. It will block access to any user which is not licensed. Make sure that you also exclude your admins from this policy. If you implement this policy you can really govern who can access the environment but requires careful planning.





CAU011-All: Block access for All users except licensed when Browser and Modern Auth Clients-v1.0		
Assignments		Access Controls
Users	Cloud Apps	Conditions
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAU011-Exclude License groups	All	Client Apps: Browser Mobile Apps and Desktop Clients
		Grant Block
		Session

5.1.4 Device

The device policies relate to the device the user is coming from. Keep in mind here that if a device which is managed but for some reason is not compliant, other policies apply targeted to non-compliant devices. In this case for example, the user will not receive any email anymore within the Outlook desktop client but can login to the Office portal with browser based restrictions.

5.1.4.1 CAD001-0365: Grant macOS access for All users when Browser and Modern Auth Clients and Compliant-v1.0

Only grant access if the macOS device is compliant. In the policy below we exclude Guests and External users, allowing them to use Client Applications to access a Teams environment hosted in your tenant. If you only want to allow browser access (either full access or restricted) you should exclude Guest and External users.

Assignments			Access Controls	
Users	Cloud Apps	Conditions	Grant	Session
All Users Except <i>Guest and External Users</i> And AAD_AA_ConAcc-Breakglass AAD_AA_CAD001-Exclude	Office 365	Client Apps: Browser Mobile Apps and Desktop Clients Device Platform: macOS	Grant Require device to be marked as compliant	

5.1.4.2 CAD002-O365: Grant Windows access for All users when Browser and Modern Auth Clients and Compliant-v1.0

Only grant access if the Windows device is compliant. In the policy below we exclude Guests and External users, allowing them to use Client Applications to access a Teams environment hosted in your tenant. If you only want to allow browser access (either full access or restricted) you should exclude Guest and External users.

Assignments			Access Controls	
Users	Cloud Apps	Conditions	Grant	Session
All Users Except <i>Guest and External Users</i> And AAD_AA_ConAcc-Breakglass AAD_AA_CAD002-Exclude	Office 365	Client Apps: Browser Mobile Apps and Desktop Clients Device Platform: Windows	Grant Require device to be marked as compliant	

5.1.4.3 CAD003-O365: Grant iOS and Android access for All users when Modern Auth Clients and ApprovedApp and Compliant-v1.0

Only grant access if the iOS or Android device is compliant or if an Approved Client App is used. In the policy below we exclude Guests and External users, allowing them to use Client Applications to





access a Teams environment hosted in your tenant. If you only want to allow browser access (either full access or restricted) you should exclude Guest and External users.

Microsoft is transitioning towards apps having a protection policy applied to eventually replace the Approved Apps functionality in some of the scenarios. For now, using this option isn't advised yet since the Microsoft Teams app is not yet supported. See my article: [Mobile Application Management for Mobile Devices with Microsoft Endpoint Manager/Intune deep dive](#)

Did you know that even if your apps are managed by another company, you can switch profiles with Microsoft Edge. So, in this case, even though the apps are managed by company A, you can switch the logged in session in the Microsoft Edge browser to company B requiring an Approved App as well. If you just want to be able to use any browser, do not include the Browser option, and the user will then receive "CAD006-O365: Session block download on unmanaged device when All users when Browser" which uses App enforced restrictions. (no download and no printing) just as on non-compliant Windows and macOS devices. If you require that web access on mobile devices should only be possible from a managed browser (Microsoft Edge) you must include the Browser in the Client App selection.

CAD003-O365: Grant iOS and Android access for All users when Modern Auth Clients and ApprovedApp or Compliant-v1.0		
Assignments		
Users	Cloud Apps	Conditions
All Users Except Guest and External Users And AAD_AA_ConAcc-Breakglass AAD_AA_CAD003-Exclude	Office 365	Client Apps: Mobile Apps and Desktop Clients Device Platform: iOS and Android
		Grant Require device to be marked as compliant or Require Approved Client App
		Session

5.1.4.4 CAD004-O365: Grant Require MFA for All users when Browser and Non-Compliant-v1.1

Require MFA if the device falls out of compliance.

CAD004-O365: Grant Require MFA for All users when Browser and Non-Compliant-v1.1		
Assignments		
Users	Cloud Apps, User Actions or Authentication Context	Conditions
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD004-Exclude	Office 365	Client Apps: Browser Filter for devices: Exclude
		Grant Require multi-factor authentication
		Session

5.1.4.5 CAD005-O365: Block access for unsupported device platforms for All users when Modern Auth Clients-v1.0

Block unsupported device platforms, like Linux and Windows Phone from accessing the environment.

CAD005-O365: Block access for unsupported device platforms for All users when Modern Auth Clients-v1.0		
Assignments		
Users	Cloud Apps	Conditions
All Users Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD005-Exclude	Office 365	Client Apps: Browser Device Platform: Any, except: Android iOS Windows macOS
		Block
		Session

Before you enable this policy, make sure that you have no "unknown" clients accessing the environment. You should check Azure AD sign-in logging as described in the article: [Microsoft is going to disable basic/legacy authentication for Exchange Online. What does that actually mean and does that impact me?](#)





5.1.4.6 CAD006-O365: Session block download on unmanaged device when All users when Browser-v1.1

This policy uses the App Enforced Restrictions, blocking download of files in OneDrive/SharePoint and Outlook Web Access. See: [Limit Access to Outlook Web Access, SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions](#).

CAD006-O365: Session block download on unmanaged device when All users when Browser-v1.1		
Assignments		
Users	Cloud Apps, User Actions or Authentication Context	Conditions
All Users	Office 365	Client Apps: Browser Filter for devices: Exclude isCompliant Equals True
Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD006-Exclude		

Access Controls		
Grant	Session	
Use App Enforced Restrictions		

5.1.4.7 CAD007-O365: Session set Sign-in Frequency for Apps for All users when Modern Auth Clients and Non-Compliant-v1.0

With this policy, you force users using Modern Authentication Clients to reauthenticate after a specified number of hours/days. I normally set this to once per 7 days.

CAD007-O365: Session set Sign-in Frequency for Apps for All users when Modern Auth Clients and Non-Compliant-v1.0		
Assignments		
Users	Cloud Apps, User Actions or Authentication Context	Conditions
All Users	Office 365	Client Apps: Mobile Apps and Filter for devices: Exclude Device Platform: iOS Android
Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD007-Exclude		

Access Controls		
Grant	Session	
Sign-in Frequency: 7 Days		

5.1.4.8 CAD008-All: Session set Sign-in Frequency for All users when Browser and Non-Compliant-v1.1

With this policy, you force users using the Browser to reauthenticate after a specified number of hours/days. I normally set this to once per day.

CAD008-All: Session set Sign-in Frequency for All users when Browser and Non-Compliant-v1.1		
Assignments		
Users	Cloud Apps, User Actions or Authentication Context	Conditions
All Users	All	Client Apps: Browser Filter for devices: Exclude isCompliant Equals True
Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD008-Exclude		

Access Controls		
Grant	Session	
Sign-in Frequency: 1 Days		





5.1.4.9 CAD009-All: Session disable browser persistence for All users when Browser and Non-Compliant-v1.1

This policy makes sure that the session is not persisted in the Browser when the browser is closed. See: [Understanding and governing reauthentication settings in Azure Active Directory](#) for more information.

CAD009-All: Session disable browser persistence for All users when Browser and Non-Compliant-v1.1				
Users	Assignments		Access Controls	
	Cloud Apps, User Actions or Authentication Context	Conditions	Grant	Session
All Users	All	Client Apps: Browser Filter for devices: Exclude isCompliant Equals True		Persistent Browser Session
Except				
AAD_AA_ConAcc-Breakglass				
AAD_AA_CAD009-Exclude				

5.1.4.10 CAD010-All: Require MFA for device join or registration when Browser and Modern Auth Clients-v1.0

This policy specifies that during Azure AD join or register operations, MFA is required

CAD010-All: Require MFA for device join or registration when Browser and Modern Auth Clients-v1.0				
Users	Assignments		Access Controls	
	Cloud Apps, User Actions or Authentication Context	Conditions	Grant	Session
All Users	User actions: Register or join devices		Grant	
Except				
AAD_AA_ConAcc-Breakglass				
AAD_AA_CAD010-Exclude				

5.1.5 Location

Location based Conditional Access policies relate to the location the user is coming from. I'm personally not a fan of excluding company locations from MFA policies. I had some cases for example where company cases were excluded but came to the conclusion that the Guest WiFi network allowing all customers used the same internet IP address, making this network trusted as well.

5.1.5.1 CAL001-All: Block specified locations for All users when Browser and Modern Auth Clients-v1.0

If your company doesn't do business in certain countries, you might as well block access from these countries. Even though there are ways to circumvent this (like using a VPN for example), it might make your security a little bit better than your neighbor.

CAL001-All: Block specified locations for All users when Browser and Modern Auth Clients-v1.0				
Users	Assignments		Access Controls	
	Cloud Apps	Conditions	Grant	Session
All Users	All	Client Apps: Browser Mobile Apps and Desktop Clients Locations: Blocked Locations	Block	
Except				
AAD_AA_ConAcc-Breakglass				
AAD_AA_CAL001-Exclude				

5.1.5.2 CAL002-All: Require MFA registration from trusted locations only for All users when Browser and Modern Auth Clients-v1.0

MFA registration, is something that you might want to allow only when the user is in a trusted location. For example, when onboarding the user.

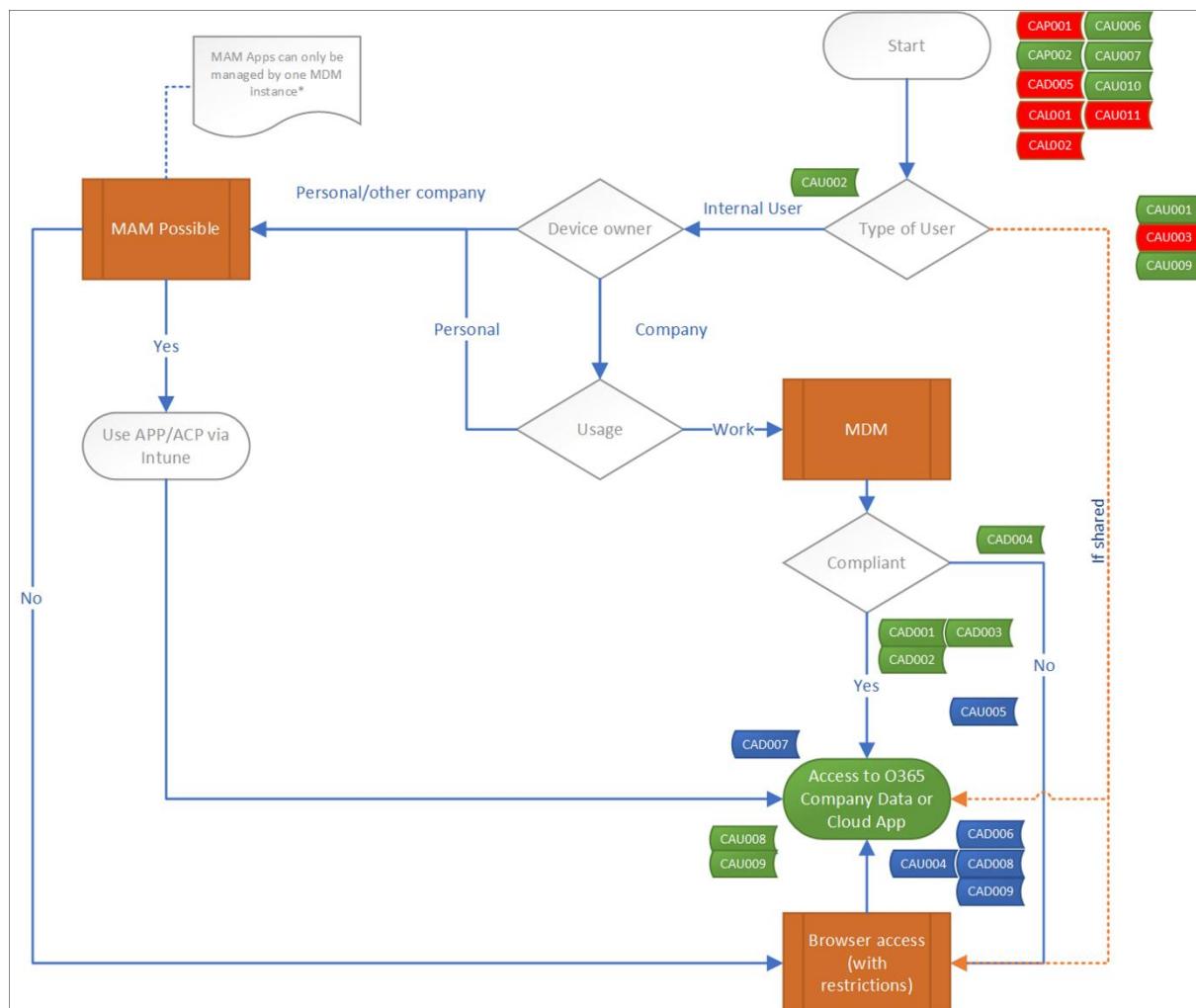




CAL002>All: Require MFA registration from trusted locations only for All users when Browser and Modern Auth Clients-v1.0		
Assignments		Access Controls
Users	Cloud Apps	Conditions
All Users	All	Client Apps: Browser Mobile Apps and Desktop Clients
Except	Actions: Register Security Information	Locations: All Locations, except Trusted Locations
AAD_AA_ConAcc-Breakglass		
AAD_AA_CAL002-Exclude		

5.1.6 Translating the functional design to the technical implementation

Below is an overview of the functional flowchart, with tags for the Conditional Access policies. Even though the position of the CA policy is not always fully correct or can be applicable in more than one scenario, it gives an idea on where the policy is applied and might help while troubleshooting.





5.2 Limit Access to Outlook Web Access and SharePoint Online and OneDrive using Conditional Access App Enforced Restrictions

One of the scenario's we can build with Conditional Access, is the scenario where we restrict access inside the web application itself. By doing so, you could for example limit the functionality of the web applications on non-managed devices, or when accessing the web application from a country where your company normally does not operate. The web applications can be configured to behave differently if the user is applicable for a Conditional Access policy where App Enforced restrictions are configured.

Within the Office 365 suite of applications, the following web applications are supported for App Enforced Restrictions:

- Outlook Web Access
- SharePoint and OneDrive

5.2.1 Configure Outlook Web Access for limited access via App Enforced Restrictions

Before you can enable Conditional Access App Enforced Restrictions you first need to enable the feature in the default OWA mailbox policy, since by default this functionality is turned off, this can be done using the [Set-OwaMailBoxPolicy](#) cmdlet as part of the [Exchange Online PowerShell module](#).

First request the current status of the OWA mailbox policy by executing the following command: Get-OwaMailBoxPolicy |select-object ConditionalAccess*. This command will return the current status of the ConditionalAccessPolicy (On or Off) and the ConditionalAccessFeatures.

If the ConditionalAccessPolicy is set to Off, you can enable the functionality which allows for restrictions when used in combination with a Conditional Access App Enforced Restriction policy. You have either the option to configure the policy in two modes:

- ReadOnly, where users can't download attachments to their local computer, and can't enable Offline Mode on non-compliant computers
- ReadOnlyPlusAttachmentsBlocked, in the ReadOnly setting viewing attachments in the browser is possible, when using this setting viewing attachments in the browser is blocked.

In this example we are going to enable the ConditionalAccessPolicy in the OWA Mailbox Policy and use the ReadOnly mode, this can be accomplished by executing the following command: Set-OwaMailboxPolicy -Identity OwaMailboxPolicy-Default -ConditionalAccessPolicy ReadOnly

After executing the command make sure that you check whether the setting was succesfull by executing the Get-OwaMailBoxPolicy |select-object ConditionalAccess* command again and check whether the ConditionalAccessPolicy is set to ReadOnly





```
Windows PowerShell
PS C:\Users\KennethvanSurksum> Get-OwaMailBoxPolicy | select-object ConditionalAccess*
ConditionalAccessPolicy ConditionalAccessFeatures
-----
Off          {}

PS C:\Users\KennethvanSurksum> Set-OwaMailBoxPolicy -Identity OwaMailboxPolicy-Default -ConditionalAccessPolicy ReadOnly
PS C:\Users\KennethvanSurksum> Get-OwaMailBoxPolicy | select-object ConditionalAccess*
ConditionalAccessPolicy ConditionalAccessFeatures
-----
ReadOnly      {Offline, AttachmentDirect FileAccessOnPrivateComputersEnabled, AttachmentDirect FileAccessOnP...
PS C:\Users\KennethvanSurksum>
```

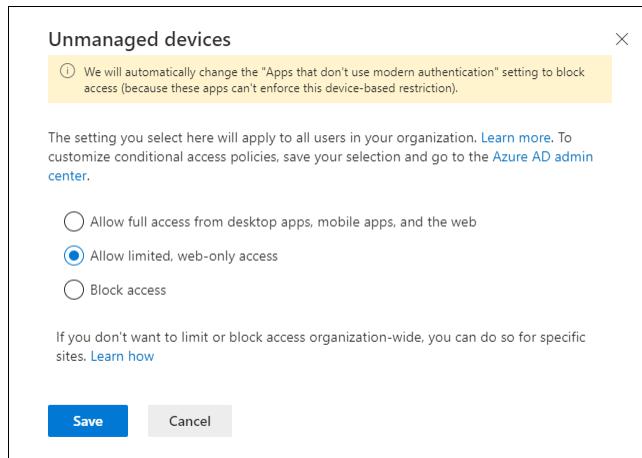
5.2.2 Configure SharePoint Online and OneDrive for limited access via App Enforced Restrictions

SharePoint Online and OneDrive can be configured in several ways. The first option is to set a Global setting which becomes effective for all SharePoint Online and OneDrive sites in your environment, and the Per site option allows you to specify options per site.

5.2.2.1 *Global settings*

The Global or organizational-wide settings can be configured from the SharePoint admin center (<https://<tenantname>-admin.sharepoint.com>). In the SharePoint Admin center select Policies | Access Control and select Unmanaged devices.

By default, for unmanaged devices the option "Allow full access from desktop apps, mobile apps, and the web" is selected, and by modifying the option to either "Allow limited, web-only access" or "Block access" you configure limited access for your whole environment.



If you configure the unmanaged devices settings, 2 new Conditional Access policies will be created. While I'm not a big fan of letting a setting like create the Conditional Access policies for you, they do provide some valuable information. I would therefore suggest to let the wizard create them, and then turn them off immediately after creation. Keep in mind that if you are playing with these, it might be that several Conditional Access policies are created (one per day, since the day when created is defined in the name)

[SharePoint admin center]Block access from apps on unmanaged devices - 2020-06-25	On	...
[SharePoint admin center]Use app-enforced Restrictions for browser access - 2020-06-25	On	...





The first one "[SharePoint admin center]Block access from apps on unmanaged devices - 2020-06-25" has the following properties

Name: [SharePoint admin center]Block access from apps on unmanaged devices - 2020-06-25

Assignments

Users and Groups: All Users

Cloud apps or actions: Office 365 SharePoint Online

Conditions

Client apps: Modern Authentication clients

Access controls

Grant: Require device to be marked as compliant OR Require Hybrid Azure AD joined device

So, to summarize this policy grants access to either Compliant Azure AD joined devices or Hybrid joined (AD joined, Azure AD registered) devices when the client supports Modern Authentication while accessing SharePoint Online.

So having clients which support Modern Authentication is crucial for this.

The second one "[SharePoint admin center] Use app-enforced Restrictions for browser access - 2020-06-25" has the following properties

Name: [SharePoint admin center]Use app-enforced Restrictions for browser access - 2020-06-25

Assignments

Users and Groups: All Users

Cloud apps or actions: Office 365 SharePoint Online

Conditions

Client apps: Browser

Access controls

Session: Use app enforced restrictions

This conditional access policy when applicable gives SharePoint online, the signal that the limited access is applicable.

5.2.2.2 *Per site settings*

For a basic environment, having these global settings might be enough, but perhaps you want to more granularly control whether you want the limited access applied to a SharePoint or OneDrive site. This can be accomplished by using PowerShell (for now, more on that later) and described in the following section "[Block or limit access to a specific SharePoint site or OneDrive](#)" of the article: "[Control access from unmanaged devices](#)" in the SharePoint online documentation.

In order to use PowerShell you must have the [SharePoint Online Management Shell](#) installed, once installed, you can connect using

Connect-SPOSERVICE -Url <https://<tenantname>-admin.sharepoint.com> which will log you in, into the Management Shell

From there you can determine the current status of a particular SharePoint site using the following command.





Get-SPOSite -Identity <https://<SharePoint online URL>/sites/<name of site or OneDrive account>> -Detailed | fl Conditional*

```
Administrator: Windows PowerShell
PS C:\Windows\system32> Get-SPOSite -identity https://insight24.sharepoint.com/sites/demo-klant1 -Detailed | fl Conditional*
ConditionalAccessPolicy : AllowFullAccess
```

As you can see, this command has a strange outcome, since on a Global level we just defined Read Only Access. Turns out that if you specify the setting on site level, this will override the global policy. So for example, you could set the option "Allow limited, web-only access" as described above in the Global policy, but define a setting to Block Access on the Site level.

You can set the policy on a site level by executing the following command: Set-SPOSite -Identity <https://<SharePoint online URL>/sites/<name of site or OneDrive account>> -ConditionalAccessPolicy <value>

```
Administrator: Windows PowerShell
PS C:\Windows\system32> Set-SPOSite -identity https://insight24.sharepoint.com/sites/demo-klant1 -ConditionalAccessPolicy BlockAccess
PS C:\Windows\system32> Get-SPOSite -identity https://insight24.sharepoint.com/sites/demo-klant1 -Detailed | fl Conditional*
ConditionalAccessPolicy : BlockAccess
```

For the ConditionalAccessPolicy parameter the following values are available:

- AllowFullAccess: The default setting
- AllowLimitedAccess: The setting allowing limited access
- BlockAccess: Blocks access

When using the AllowLimitedAccess option, you can supply additional parameters to further define the behavior, as detailed in "[Advanced Configurations](#)", for example you can provide the option -LimitedAccessType OtherFiles after the -ConditionalAccess AllowLimitedAccess parameter to allow users to download files that can't be previewed, such as .ZIP files.

If you must specify this for a lot of SharePoint sites, you can of course automate these settings, but wouldn't it be nice if we could enable this from the GUI while creating the SharePoint site, or while the SharePoint site is created as part of a Teams environment.

5.2.2.3 Using sensitivity labels for containers

What if you want more granularity and want to decide on a per SharePoint site basis whether or not these App Enforced Restrictions should be applicable?

Luckily, there are options.

5.2.2.3.1 Option 1: Block or limit access to a specific SharePoint site or OneDrive

This option is explained in the following article: [Control access from unmanaged devices](#), the article explains that by using PowerShell you can limit access by using the Set-SPOSite commandlet.





Set-SPOSite -Identity <https://<SharePoint online URL>/sites/<name of site or OneDrive account>> - ConditionalAccessPolicy AllowLimitedAccess

When using this option, you must remove the global setting, since setting another setting on a subsite only works when its less restrictive. If you would for example set the global policy to "Allow limited, web-only access" and use the Set-SPOSite commandlet to set the Conditional Access Policy for a specific site to "Allow full access from desktop apps, mobile apps, and the web" using the AllowFullAccess parameter, the access will still be limited. If however you would set the Conditional Access Policy for a specific site to "Block Access" using the BlockAccess parameter, the access to the site will be blocked.

Access Denied

Due to organizational policies, you can't access this resource from this untrusted device.

Here are a few ideas:

[Please contact your organization.](#)

If this problem persists, contact your support team and include these technical details:

Correlation ID: 45b7939f-9009-b000-32ac-1ba8b869e48c
Date and Time: 3-12-2020 07:45:18
User: fleuhman@emshelden.nl
Issue Type: User has encountered a policy issue.

Using this method, has major disadvantages, since you have to execute the necessary PowerShell command for each SharePoint and OneDrive after its created. This can easily be forgotten and can lead to inconsistency.

5.2.2.3.2 Option 2: Use Sensitivity labels for Containers

There is far more to tell when it comes to Sensitivity labels then explained in this blogpost. For this blogpost we are going to make use of Sensitivity labels for contains, which can be used to define certain settings when creating a Teams or SharePoint environment.

By using sensitivity labels for containers we can control the following settings:

- Privacy and external user access settings
 - Use the label to determine whether the site privacy is set to Public, Private or None
 - Define whether Microsoft 365 Group owners can add Guest users to the group.
- Device access and external sharing settings
 - You can determine if external sharing settings already on the site will be replaced or respected.
 - You can determine the access from unmanaged devices (same options as on global level - Full, Limited and Block)





So, by defining Sensitivity labels for containers we can actually determine the access from unmanaged devices setting that will be used when the Conditional Access policy which enforces the App Enforced Restrictions will be hit.

The settings of this policy is explained in my article: "[Conditional Access demystified: My recommended default set of policies](#)", the name of the policy providing this functionality is called: "CAD006-O365: Session block download on unmanaged device when All users when Browser-v1.0"

CAD006-O365: Session block download on unmanaged device when All users when Browser-v1.0			
Assignments			Access Controls
Users	Cloud Apps	Conditions	Grant
All Users	Office 365	Client Apps: Browser	
Except AAD_AA_ConAcc-Breakglass AAD_AA_CAD006-Exclude		Device state: All except Device marked as Compliant	Session Use App Enforced Restrictions

After creating the sensitivity labels, you can use them for each new Teams/SharePoint site created, and based on the defined Sensitivity label the correct access from unmanaged devices setting will be applied.

This is not a perfect solution, since it will not solve the issue for all SharePoint sites already created, and you still must build a solution for OneDrive sites which are created automatically and do not have the option to define a "default" sensitivity label at creation.

Let's go a little bit more into detail in how to build this from scratch, let's walk through the steps.

Step 1: Enable sensitivity labels for containers.

By default, support for sensitivity labels for containers is not enabled, you can easily determine this by creating a new sensitivity label or by trying to modify an existing one. If the option to select Group & sites is greyed out, you first must execute some steps to enable this functionality.





Edit sensitivity label

Define the scope for this label

Labels can be applied directly to files, emails, containers like SharePoint sites and Teams, and more. Let us know where you want this label to be used so you can configure the applicable protection settings. [Learn more about label scopes](#)

Files & emails
Configure encryption and content marking settings to protect labeled emails and Office files. Also define auto-labeling conditions to automatically apply this label to sensitive content in Office, files in Azure, and more.

Groups & sites
Configure privacy, access control, and other settings to protect labeled Teams, Microsoft 365 Groups, and SharePoint sites.
(ⓘ To apply sensitivity labels to Teams, SharePoint sites, and Microsoft 365 Groups, you must first [complete these steps](#) to enable the feature.

Azure Purview assets (preview)
Apply label to assets in Azure Purview, including SQL columns, files in Azure Blob Storage, and more.
(ⓘ To apply this label to Azure Purview assets, you must first turn on labeling for Azure Purview. You can do this from the Labels page. [Learn more about labeling for Azure Purview](#)

[Back](#) [Next](#) [Cancel](#) [Need help?](#) [Give feedback](#)

The following procedure is explained in the following article: [How to enable sensitivity labels for containers and synchronize labels](#) and one of the first steps is to enable the feature to apply sensitivity labels to groups as explained in this article: [Assign sensitivity labels to Microsoft 365 groups in Azure Active Directory](#)

The first thing we need to do, is to import the AzureADPreview module using the import-module AzureADPreview and connect using the Connect-AzureAD commandlet. Make sure that you use the Connect-AzureAD commandlet from the Azure AD Preview module by putting AzureADPreview\ in front of it. Once connected verify if group settings have been set for the Azure AD organization. If no group settings are applied, you'll get the same error as in the picture below.

```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> import-module AzureADPreview
PS C:\WINDOWS\system32> AzureADPreview\Connect-AzureAD
Account Environment TenantId TenantDomain AccountType
-----
admin@M365x102715.onmicrosoft.com AzureCloud 126de6ea-ffb9-445e-a91f-2d49eaf150de M365x102715.onmicrosoft.com User

PS C:\WINDOWS\system32> $Setting = Get-AzureADDirectorySetting -Id (Get-AzureADDirectorySetting | where -Property DisplayName -Value "Group.Unified" -EQ).id
Get-AzureADDirectorySetting : Cannot bind argument to parameter 'Id' because it is null.
At line:1 char:44
+ ... Setting -Id (Get-AzureADDirectorySetting | where -Property DisplayName ...
+
+ CategoryInfo          : InvalidData: () [Get-AzureADDirectorySetting], ParameterBindingValidationException
+ FullyQualifiedErrorId : ParameterArgumentValidationErrorNullNotAllowed,Microsoft.Open.MSGraphBeta.PowerShell.Get
DirectorySetting
PS C:\WINDOWS\system32>
```

If this is the case you first have to create the settings, as explained in the following article: [Azure Active Directory cmdlets for configuring group settings](#)





```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> Get-AzureADDirectorySettingTemplate
Id                               DisplayName          Description
--                               -----
08d542b9-071f-4e16-94b0-74abb372e3d9 Group.Unified.Guest      Settings for a specific Unified Group
4bc7f740-180e-4586-adb6-38b2e9024e6b Application           ...
898f1161-d651-43d1-805c-3b0b388a9fc2 Custom Policy Settings ...
80661d51-be2f-4d46-9713-98a2fcac5bc Prohibited Names Settings ...
aad3907d-1d1a-448b-b3ef-7bf7f63db63b Prohibited Names Restricted Settings ...
5cf42378-d67d-4f36-ba46-e8b86229381d Password Rule Settings ...
62375ab9-6b52-47ed-826b-58e47e0e304b Group.Unified           ...
dfffd5d46-495d-40a9-8e21-954ff55e198a Consent Policy Settings ...

PS C:\WINDOWS\system32> $TemplateId = (Get-AzureADDirectorySettingTemplate | where { $_.DisplayName -eq "Group.Unified" }).Id
>> $Template = Get-AzureADDirectorySettingTemplate | where -Property Id -Value $TemplateId -EQ
PS C:\WINDOWS\system32> $Setting = $Template.CreateDirectorySetting()
PS C:\WINDOWS\system32> $Setting["UsageGuidelinesUrl"] = "https://guideline.insight24.nl"
PS C:\WINDOWS\system32> New-AzureADDirectorySetting -DirectorySetting $Setting
Id                               DisplayName TemplateId          Values
--                               -----
c7d92f4b-003f-45f7-b3b0-531d5a5dfbab       62375ab9-6b52-47ed-826b-58e47e0e304b {class SettingValue {...

PS C:\WINDOWS\system32> $Setting.Values
Name                Value
---                ---
EnableMIPLabels     False
CustomBlockedWordsList
EnableMSStandardBlockedWords  False
ClassificationDescriptions
DefaultClassification
PrefixSuffixNamingRequirement
AllowGuestsToBeGroupOwner  False
AllowGuestsToAccessGroups True
GuestUsageGuidelinesUrl
GroupCreationAllowedGroupId
AllowToAddGuests      True
UsageGuidelinesUrl   https://guideline.insight24.nl
ClassificationList
EnableGroupCreation  True

PS C:\WINDOWS\system32>
```

After performing these steps, you can continue with enabling the Microsoft Information Protection labels functionality as shown in the figure below. You can see that the EnableMIPLabels value is set to True.





```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> $Setting = Get-AzureADDirectorySetting -Id (Get-AzureADDirectorySetting | where -Property DisplayName -Value "Group.Unified" -EQ).id
PS C:\WINDOWS\system32> $Setting.Values

Name          Value
----          -----
EnableMIPLabels      False
CustomBlockedWordsList
EnableMSStandardBlockedWords  False
ClassificationDescriptions
DefaultClassification
PrefixSuffixNamingRequirement
AllowGuestsToBeGroupOwner  False
AllowGuestsToAccessGroups  True
GuestUsageGuidelinesUrl
GroupCreationAllowedGroupId
AllowToAddGuests        True
UsageGuidelinesUrl     https://guideline.insight24.nl
ClassificationList
EnableGroupCreation    True

PS C:\WINDOWS\system32> $Setting["EnableMIPLabels"] = "True"
PS C:\WINDOWS\system32> Set-AzureADDirectorySetting -Id $Setting.Id -DirectorySetting $Setting
PS C:\WINDOWS\system32> $Setting.Values

Name          Value
----          -----
EnableMIPLabels      True
CustomBlockedWordsList
EnableMSStandardBlockedWords  False
ClassificationDescriptions
DefaultClassification
PrefixSuffixNamingRequirement
AllowGuestsToBeGroupOwner  False
AllowGuestsToAccessGroups  True
GuestUsageGuidelinesUrl
GroupCreationAllowedGroupId
AllowToAddGuests        True
UsageGuidelinesUrl     https://guideline.insight24.nl
ClassificationList
EnableGroupCreation    True

PS C:\WINDOWS\system32>
```

After this is done, you must synchronize your sensitivity labels to Azure AD. You can do this by connecting to Security & Compliance PowerShell using the Connect-IPPSSession commandlet

```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> connect-IPPSSession
WARNING: Your connection has been redirected to the following URI:
"https://eur02b.ps.compliance.protection.outlook.com/Powershell-LiveId?BasicAuthToOAuthConversion=true&HideBannerMessage=true;PSVersion=5.1.19041.610"
PS C:\WINDOWS\system32> Execute-AzureAdLabelSync
PS C:\WINDOWS\system32>
```

Once finished, the result should be that you are able to specify the "Groups & Sites" option when modifying an existing Sensitivity label or creating a new one.





Edit sensitivity label

Scope

- Name & description
- Scope
- Files & emails
- Groups & sites
- Azure Purview assets (preview)
- Finish

Define the scope for this label

Labels can be applied directly to files, emails, containers like SharePoint sites and Teams, and more. Let us know where you want this label to be used so you can configure the applicable protection settings. [Learn more about label scopes](#)

Files & emails

Configure encryption and content marking settings to protect labeled emails and Office files. Also define auto-labeling conditions to automatically apply this label to sensitive content in Office, files in Azure, and more.

Groups & sites

Configure privacy, access control, and other settings to protect labeled Teams, Microsoft 365 Groups, and SharePoint sites.

Azure Purview assets (preview)

Apply label to assets in Azure Purview, including SQL columns, files in Azure Blob Storage, and more.

To apply this label to Azure Purview assets, you must first turn on labeling for Azure Purview. You can do this from the Labels page. [Learn more about labeling for Azure Purview](#)

Back Next Cancel Need help? Give feedback

Step 2: Create or Modify your Sensitivity Labels.

So, now that we have the functionality available, we can define our settings. In this blogpost I used the following settings

Label name	Privacy and external user access settings	Device access and external sharing settings
Public	Privacy: Public External user access: Enabled	Content can be shared with: Anyone. Users can share files and folders using links that don't require sign-in Access from unmanaged devices: Allow full access from desktop apps, mobile apps, and the web
General	Privacy: Private External user access: Enabled	Content can be shared with: New and existing guests. Guests must sign in or provide a verification code Access from unmanaged devices: Allow full access from desktop apps, mobile apps, and the web
Confidential	Privacy: Private External user access: Enabled	Content can be shared with: New and existing guests. Guests must sign in or provide a verification code Access from unmanaged devices: Allow limited, web only access
Highly Confidential	Privacy: Private External user access: Disable	Content can be shared with: Only people in your organization. No external sharing allowed Access from unmanaged devices: Block access

Next some figures showing the configuration of the Groups & Sites setting of my Confidential label





Edit sensitivity label

- Name & description
- Scope
- Files & emails
- Groups & sites
- Azure Purview assets (preview)
- Finish

Define protection settings for groups and sites

These settings apply to teams, groups, and sites that have this label applied. They don't apply directly to the files stored in those containers. [Learn more about these settings](#)

Privacy and external user access settings

Control the level of access that internal and external users will have to labeled teams and Microsoft 365 Groups.

Device access and external sharing settings

Control external sharing and unmanaged device access for labeled SharePoint sites.

Back

Next

Cancel

Need help?

Give feedback



Edit sensitivity label

- Name & description
- Scope
- Files & emails
- Groups & sites
- Azure Purview assets (preview)
- Finish

Define protection settings for groups and sites

These settings apply to teams, groups, and sites that have this label applied. They don't apply directly to the files stored in those containers. [Learn more about these settings](#)

Privacy and external user access settings

Control the level of access that internal and external users will have to labeled teams and Microsoft 365 Groups.

Device access and external sharing settings

Control external sharing and unmanaged device access for labeled SharePoint sites.

Back

Next

Cancel

Need help?

Give feedback





Edit sensitivity label

- Name & description
- Scope
- Files & emails
- Groups & sites
- Privacy & external user access
 - Public. Anyone in your organization can access the group or team (including content) and add members.
 - Private. Only team owners and members can access the group or team, and only owners can add members.
 - None. Team and group members can set the privacy settings themselves.
- External sharing & device access
- Azure Purview assets (preview)
- Finish

Define privacy and external user access settings

Control the level of access that internal and external users will have to labeled teams and Microsoft 365 Groups.

Privacy

These options apply to all Microsoft 365 Groups and teams that have this label applied. When applied, these settings will replace any existing privacy settings for the team or group. If the label is removed, users can change it again.

- Public. Anyone in your organization can access the group or team (including content) and add members.
 Private. Only team owners and members can access the group or team, and only owners can add members.
 None. Team and group members can set the privacy settings themselves.

External user access

Let Microsoft 365 Group owners add people outside your organization to the group as guests. [Learn about guest access](#)

[Back](#) [Next](#)

[Cancel](#)

[Need help?](#) [Give feedback](#) [▼](#)

Edit sensitivity label

- Name & description
- Scope
- Files & emails
- Groups & sites
- Privacy & external user access
- External sharing & device access
- Azure Purview assets (preview)
- Finish

Define external sharing and device access settings

Control who can share SharePoint content with people outside your organization and decide whether users can access labeled sites from unmanaged devices.

Control external sharing from labeled SharePoint sites

When this label is applied to a SharePoint site, these settings will replace existing external sharing settings configured for the site.

- Content can be shared with
- Anyone. Users can share files and folders using links that don't require sign-in. [○](#)
 New and existing guests. Guests must sign in or provide a verification code. [○](#)
 Existing guests. Only guests in your organization's directory. [○](#)
 Only people in your organization. No external sharing allowed.

Access from unmanaged devices

Determine whether users can access SharePoint sites from unmanaged devices (which are devices that aren't hybrid Azure AD joined or enrolled in Intune).

[○](#) For this setting to work you must also configure the SharePoint feature that blocks or limits access to SharePoint files from unmanaged devices. [Learn more](#)

- Allow full access from desktop apps, mobile apps, and the web
 Allow limited, web-only access [○](#)
 Block access [○](#)

[Back](#) [Next](#)

[Cancel](#)

[Need help?](#) [Give feedback](#) [▼](#)

So now that the labels have been created, you will have the Sensitivity label options available when creating SharePoint and Teams environments as detailed in the slideshow below.

Note: It can take a while before you are able to use the sensitivity labels.





What kind of team will this be?

Sensitivity [Learn more](#)

Confidential

Teams with this sensitivity must be private.

Privacy

Private
People need permission to join

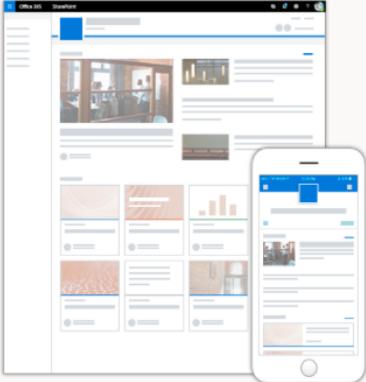
Public
Anyone in your org can join

Org-wide
Everyone in your organization automatically joins

< Back

Get a team site connected to Microsoft 365 Groups

Use this design to collaborate with your team. Share documents, track events in a shared calendar, and manage project tasks.



Site name
The site name is available.

Group email address
The group alias is available.

Site address
https://m365x102715.sharepoint.com/sites/SharePointSite
The site address is available.

Site description

Sensitivity [○](#)

Privacy settings
Private - only members can access this site

Select a language
Select the default site language for your site. You can't change this later.

Usage guidelines

Next **Cancel**





← X

Edit sensitivity setting

Select the sensitivity level you want to apply to this site. For more info about these labels, or to create a new one, go to the Security Center.

Public
Public usage

General
This is the General label

Confidential
This is the Confidential label

High Confidentiality
High Confidentiality

None

[Save](#) [Cancel](#)

Contoso Electronics SharePoint

SharePointSite Private group (Confidential)

5 Search this site

Published Edit

1 member

Home + New Page details

News + Add v Keep your team updated with news on your team site From the site home page you'll be able to quickly author a news post – a status update, trip report, or... Add News

Quick links Learn about a team site Learn how to add a page

Documents See all + New ... All Documents

Activity

SharePointSite View and share files Get organized SharePointSite Owners Collaborate on content with your team. Use lists to keep team activities organized. + Add a list

Drag files here

Microsoft Teams Add Microsoft Teams to collaborate in real-time and share files directly with Microsoft 365 with your team Add Microsoft Teams

5.2.2.3.3 How to provide a Sensitivity label to your already existing Teams and SharePoint sites?

So, now that we have configured the sensitivity labels, and can use them to create new Teams or SharePoint sites, how can we handle that our current Teams and SharePoint sites are labelled as well?

For this we need PowerShell, as explained in the following article: [Use PowerShell to apply a sensitivity label to multiple sites](#)





Make sure that you have connected to SharePoint Online and to the Security & Compliance Center PowerShell environments by using the Connect-SPOService and Connect-IPPSession commandlets. Retrieve the GUID of the label that you want to apply to all your existing sites using the Get-Label | ft Name, Guid command. Make sure to put the ID in a variable and enumerate the SharePoint sites by using a generic string representing your tenant. In my case this is "M365x102715"

```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> $Id = [GUID]("8f9825d0-12d9-44b2-bb01-85bf2799fa88")
PS C:\WINDOWS\system32> $sites = Get-SPOSite -IncludePersonalSite $true -Limit all -Filter "Url -like 'M365x102715'"
PS C:\WINDOWS\system32> write-output $sites

Url                                         Owner
---                                         -----
https://m365x102715-my.sharepoint.com/personal/christie_m365x102715_onmicrosoft_com christie@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/SharePointSite
https://m365x102715.sharepoint.com/sites/contosoteam
https://m365x102715.sharepoint.com/sites/GlobalMarketing
https://m365x102715.sharepoint.com/sites/SalesAndMarketing
https://m365x102715-my.sharepoint.com/personal/admin_m365x102715_onmicrosoft_com admin@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/Communications
https://m365x102715.sharepoint.com/sites/Contoso
https://m365x102715-my.sharepoint.com/
https://m365x102715-my.sharepoint.com/personal/alexw_m365x102715_onmicrosoft_com alexw@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/GlobalSales
https://m365x102715.sharepoint.com/sites/leadership-connection
https://m365x102715.sharepoint.com/sites/operations
https://m365x102715.sharepoint.com/sites/salesbestpractices
https://m365x102715-my.sharepoint.com/personal/lynner_m365x102715_onmicrosoft_com lynner@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/portals/hub
https://m365x102715.sharepoint.com/sites/askhr
https://m365x102715.sharepoint.com/
https://m365x102715.sharepoint.com/sites/parentsofcontoso
https://m365x102715.sharepoint.com/sites/ReviewCenterForRetention
https://m365x102715-my.sharepoint.com/personal/diegos_m365x102715_onmicrosoft_com diegos@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/ThePerspective
https://m365x102715.sharepoint.com/portals/Community
https://m365x102715.sharepoint.com/sites/CommercialLending
https://m365x102715.sharepoint.com/sites/newemployeeonboarding
https://m365x102715-my.sharepoint.com/personal/adelev_m365x102715_onmicrosoft_com adelev@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/pradeepg_m365x102715_onmicrosoft_com pradeepg@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/benefits
https://m365x102715.sharepoint.com/sites/ceoconnection
https://m365x102715-my.sharepoint.com/personal/irvins_m365x102715_onmicrosoft_com irvins@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/debrab_m365x102715_onmicrosoft_com debrab@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/ContosoBrand
https://m365x102715-my.sharepoint.com/personal/jonis_m365x102715_onmicrosoft_com jonis@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/alland_m365x102715_onmicrosoft_com alland@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/pattif_m365x102715_onmicrosoft_com pattif@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/nestorw_m365x102715_onmicrosoft_com nestorw@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/SOCTeam
https://m365x102715.sharepoint.com/sites/leadership
https://m365x102715.sharepoint.com/sites/contosolife
https://m365x102715-my.sharepoint.com/personal/meganb_m365x102715_onmicrosoft_com meganb@m365x102715.onmicrosoft.com
https://m365x102715-my.sharepoint.com/personal/miriang_m365x102715_onmicrosoft_com miriang@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/ContosoWorks
https://m365x102715.sharepoint.com/sites/ContosoNews
https://m365x102715-my.sharepoint.com/personal/leeg_m365x102715_onmicrosoft_com leeg@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/FlySafeConference
https://m365x102715.sharepoint.com/search
https://m365x102715-my.sharepoint.com/personal/lidiah_m365x102715_onmicrosoft_com lidiah@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/safety
https://m365x102715-my.sharepoint.com/personal/gradya_m365x102715_onmicrosoft_com gradya@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/Mark8ProjectTeam
https://m365x102715.sharepoint.com/sites/SalesPlanning
https://m365x102715.sharepoint.com/sites/USSales
https://m365x102715-my.sharepoint.com/personal/johannal_m365x102715_onmicrosoft_com johannal@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/give
https://m365x102715.sharepoint.com/sites/Retail
https://m365x102715.sharepoint.com/sites/droneproducttraining
https://m365x102715-my.sharepoint.com/personal/isaiah1_m365x102715_onmicrosoft_com isaiah1@m365x102715.onmicrosoft.com
https://m365x102715.sharepoint.com/sites/RetailOperations
https://m365x102715.sharepoint.com/sites/productsupport
https://m365x102715.sharepoint.com/sites/office365adoption
https://m365x102715.sharepoint.com/sites/DigitalInitiativePublicRelations
```

Now that we have enumerated all SharePoint sites (including OneDrive sites) we can apply the label we want, In my case I have chosen to use the Confidential label, so that by default I provide limited access and can use the GUI to make exceptions.





```
Administrator: Windows PowerShell
PS C:\WINDOWS\system32> $sites | ForEach-Object {Set-SPOTenant $_.url -SensitivityLabel $Id}
Set-SPOTenant : The property SensitivityLabel2 cannot be set on the MySite host.
At line:1 char:26
+ $sites | ForEach-Object {Set-SPOTenant $_.url -SensitivityLabel $Id}
+ ~~~~~
+     + CategoryInfo          : NotSpecified: (:) [Set-SPOTenant], ServerException
+     + FullyQualifiedErrorId : Microsoft.SharePoint.Client.ServerException,Microsoft.Online.SharePoint.PowerShell.SetTe
nant
PS C:\WINDOWS\system32>
```

Apparently, you cannot set a sensitivity label on the MySite host, which is the <https://m365x102715-my.sharepoint.com/> URL in my case. But this can be ignored.

The end result is that all the SharePoint sites, and OneDrive sites will have the Confidential sensitivity label applied on the container level

Site name	URL	Storage used (GB)	Date created	Sensitivity	External sharing
A&R HR	...sites/askhr	0.00	10/29/20, 10:02 AM	Confidential	On
Benefits	...sites/benefits	0.03	10/29/20, 11:47 PM	Confidential	On
CEO Connection	...sites/ceoconnection	0.00	10/29/20, 10:02 AM	Confidential	On
Commercial Lending	...sites/CommercialLending	0.01	10/31/20, 8:33 AM	Confidential	On
Communications	...sites/Communications	0.60	10/29/20, 10:25 PM	Confidential	On
Contoso	...sites/Contoso	0.00	10/29/20, 10:17 AM	Confidential	On
Contoso Brand	...sites/ContosoBrand	0.02	10/29/20, 3:00 PM	Confidential	On
Contoso Life	...sites/contosolife	0.00	10/29/20, 10:03 AM	Confidential	On
Contoso News	...sites/ContosoNews	0.08	10/29/20, 2:46 PM	Confidential	On
Contoso Team	...sites/contosoteam	0.04	10/29/20, 10:13 AM	Confidential	On
Contoso Works	...sites/ContosoWorks	0.02	10/29/20, 11:59 PM	Confidential	On
Digital Initiative Public Relations	...sites/DigitalInitiativePublicRelations	0.23	10/29/20, 10:13 AM	Confidential	On
Drone workshop	...sites/droneproductctraining	0.01	10/30/20, 1:22 AM	Confidential	On
Fly Safe Conference	...sites/FlySafeConference	0.02	10/30/20, 12:24 AM	Confidential	On
Give	...sites/give	0.03	10/29/20, 9:50 PM	Confidential	On

Keep in mind though that you have to create a procedure from now on to make sure that the Sensitivity label gets applied to newly created OneDrive sites. Unfortunately I haven't found a way yet to set a default Sensitivity label for newly created OneDrive sites.

5.2.2.4 Policy behavior within Outlook Web Access

When the Conditional Access policy is applicable, the user accessing Outlook Web Access experiences the following behavior.

Your organization doesn't allow you to download or print attachments from this device or browser. You can still view attachments in your browser. For more information, contact your IT administrator. [Learn More](#)

5.2.2.5 Policy behavior within SharePoint Online and OneDrive

When the Conditional Access policy is applicable, the user accessing SharePoint or OneDrive experiences the following behavior.





INSIGHT24 OneDrive

Your organization doesn't allow you to download, print, or sync using this device. To use these actions, use a device that's joined to a domain or marked compliant by Intune. For help, contact your IT department. [More info.](#)

Ferry Kuhlman

Open Share Copy link Delete Rename Automate Sort 1 selected

My files

Recent Shared Discover Recycle bin Shared libraries

Demo - Klant1 Create shared library

Completed the Microsoft 365 ...

Modified Name Modified By File size Sharing

Open Ferry Kuhlman 305 KB Private

Preview Share Copy link Manage access

Delete Rename Automate Details

Get the OneDrive apps Return to classic OneDrive





6 Testing and troubleshooting conditional access

In this chapter we will go into more detail on where we can find information which can help us to test and troubleshoot Conditional Access policies.

6.1 What if tool

You can find the What if tooling by clicking on the What If icon on the Conditional Access policy overview page.

The screenshot shows the 'What If' tool interface. At the top, there's a header with 'What If' and 'Policies'. Below it is a 'Info' section with a link to 'Learn more'. The main area is titled 'Test the impact of conditional access on a user when signing in under certain conditions.' It includes sections for 'User' (0 users selected), 'Cloud apps or actions' (Any cloud app), 'IP address' (Enter IP address (ex: 40.77.182.32)), 'Country' (Select country...), 'Device platform' (Select device platform...), 'Client apps (preview)' (Select a client app...), 'Device state (preview)' (Select device state...), and 'Sign-in risk' (Select sign-in risk...). At the bottom are 'What If' and 'Reset' buttons.

With the What if tooling you can determine which policies are applicable for a certain scenario. If you run the tooling it will give you an overview of which policies will apply, and which policies will not apply including the condition that has not been met.

A possible outcome can be:

Evaluation result

⚠️ You have one or more classic policies configured. This includes policies in enabled or disabled state. Click here to view policies.

Policies that will apply	Policies that will not apply
Baseline policy: Block legacy authentication (Preview)	Block access
I24 - Browser Access via MFA	Require multi-factor authentication
I24 - Block legacy authentication Exchange Online	Block access
I24 - Block Active Sync Exchange Online	Block access
I24 - Block login from certain countries	Block access





There are some things the What if tooling is not capable of displaying though, that is that the effective outcome for the user, take for example the outcome of the example above - I'm still granted access to the cloud apps even though some "Block access" controls apply.

6.2 Report-only Mode

By enabling the Report-only mode the conditional access is evaluated on the client instead of enforced. By using the Azure AD sign-in logging functionality we can then determine the expected behavior of the Conditional Access Policy. This can be done in two ways:

1. Using Azure Active Directory Sign-in logging

Go to the Azure AD administration portal | Monitoring | Sign-ins and select one of the listed sign-ins. Once selected, within the sign-in logging, a tab titled: "Report-only" is available. Here you can see, in this example that the "I24 – Accept User Terms" conditional access policy reports the result: "Report-only: User action required"

Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only (Preview)	Additional Details
Policy Name	↑↓	Grant Controls	↑↓	Session Controls	↑↓	Result
I24 - Accept User Terms		I24 Terms of Use				Report-only: User action required
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.						

2. Using Azure AD Workbooks

Using Azure AD workbooks requires that you have setup Log Analytics and forward your Azure AD sign-in logging to a Log Analytics Workspace.

Conditional Access Insights (Preview)

Select one or more conditional access policies to evaluate their impact. To modify or enforce a policy visit the Conditional Access blade. Submit public preview feedback [here](#).

Conditional Access policy: I24 - Accept User Terms - (R...▼) Time range ⓘ : Last 24 hours ▼ User ⓘ : All users ▼ App ⓘ : All apps ▼ Data view ⓘ : users ▼

To save your parameter selections for next time, save a copy of this workbook.

Impact Summary





6.3 Azure Active Directory sign-ins logging

Once the policy is implemented you can use sign-ins logging from Azure Active Directory. Sign-ins logging is available under the monitoring section of Azure Active Directory, in the overview you can see all the sign ins and once a sign in is selected you can find more information about the circumstances under which the sign-in took place. On the Conditional Access tab, you can find all the Conditional Access related information.

Below is an example of the outcome. On the basic page you can see the user involved, the date and time the sign in took place and what client app (Chrome browser) was used to access the cloud app (in this case Office 365 Exchange Online).

Details					
Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only
Date	5/19/2020, 7:45:09 PM		User	Kenneth van Surkum	Token issuer type Azure AD
Request ID	3156f757-85c6-4992-8f54-05cb10f69100		Username	ksurkum@insight24.nl	Token issuer name
Correlation ID	64536636-5f3a-4848-ac99-154e1c927fe4		User ID	609b2486-922b-456b-aee5-63cec02f0873	Latency 409ms
Status	Success		Alternate sign-in name		User agent Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.56 Safari/537.36 Edg/83.0.478.33
			Application	Office 365 Exchange Online	
			Application ID	00000002-0000-0ff1-ce00-000000000000	
			Resource	Office 365 Exchange Online	
			Resource ID	00000002-0000-0ff1-ce00-000000000000	
			Client app	Browser	

On the tab Conditional Access you can see which policies are applied for this login, whether the policy blocked access or denied access and what result was.

Details					
Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only
Policy Name	↑↓	Grant Controls	↑↓	Session Controls	↑↓ Result
I24 - Browser Access via MFA		require multi-factor authentication			Success
I24 - Sign-in frequency				persistent browser session, sign-in frequency	Success
I24 - Require MFA for Office Apps		require multi-factor authentication			Success
EAS					Disabled
MAM for EXO and SPO					Disabled
I24 - Block legacy authentication Exchange Online		block			Not Applied
I24 - Block Active Sync Exchange Online		block			Not Applied
I24 - Block login from certain countries		block			Not Applied
I24 - Block All (Safety Measure)		block			Not Applied
I24 - MFA for Intune Enrollment		require multi-factor authentication			Not Applied

If you click on one of the Conditional Access policies, a new pane will pop up giving you additional information about this specific Conditional Access Policy

Policy details (Preview)

Policy: I24 - Browser Access via MFA
Policy state: Enabled
Result: Success

Assignments

User	Kenneth van Surkum	<input checked="" type="checkbox"/> Satisfied
Application	Office 365 Exchange Online	<input checked="" type="checkbox"/> Satisfied

Conditions

Sign-in risk	None	<input type="radio"/> Not configured
Device Platform	Windows 10	<input type="radio"/> Not configured
Location		<input checked="" type="checkbox"/> Satisfied
Client app	Browser	<input checked="" type="checkbox"/> Satisfied
Device state	Compliant Azure AD joined	<input type="radio"/> Not configured

Access controls

Grant Controls	<input checked="" type="checkbox"/> Satisfied
----------------	---

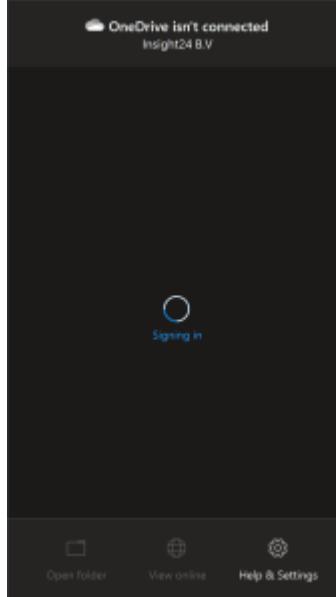




6.4 OneDrive client sign-in issues due to Conditional Access policies in Azure AD tenant where you are a guest user

This is an example of what can happen to Guest users, If you configure your own Conditional Access policies but don't take your Guest users into account.

The issue I encountered was related to the fact that I couldn't sign-in into the OneDrive client anymore. When you are not able to sign-in, you cannot open documents which are cloud hosted for example, and of course changes in files are not synchronized.



So in order to troubleshoot the issue, I started digging into the Azure AD sign-in logs, where I found the following interesting entry.

Activity Details: Sign-ins					
	Basic info	Location	Device info	Authentication Details	Conditional Access
Date	9/10/2021, 11:26:47 AM			User	Kenneth van Surksum
Request ID	172f8806-54d1-4751-8a97-7c06037b4300			Username	[REDACTED]
Correlation ID	bf70bf2-dc00-4797-aa0a-3fdb475763f5			User ID	609b2486-922b-456b-aee5-63cec02f0873
Authentication requirement	Single-factor authentication			Sign-in identifier	
Status	Failure			User type	Guest
Continuous access evaluation	No			Cross tenant access type	B2B collaboration
Sign-in error code	53000			Application	OneDrive SyncEngine
Failure reason	Device is not in required device state: (state). Conditional Access policy requires a compliant device, and the device is not compliant. The user must enroll their device with an approved MDM provider like Intune.			Application ID	ab9b8c07-8f02-4f72-87fa-80105867a763
				Resource	Office 365 SharePoint Online
				Resource ID	00000003-0000-0ff1-ce00-000000000000
				Resource tenant ID	[REDACTED]
				Home tenant ID	[REDACTED]
				Client app	
Token issuer type	Azure AD				
Token issuer name					
Latency	127ms				
Flagged for review	No				
User agent	Microsoft SkyDriveSync 21.160.0808.0002 ship; Windows NT 10.0 (19043)				





The first thing I noticed is that the User Type property was set to Guest, while normally it states "Member". It kept me wondering why my account was a Guest user and then it all became clear. For a project which I was working on at another customer, the customer asked me for my IP address so that they could allow that IP address to access the Teams environment being created. I have setup a sync of one of the SharePoint folder within my OneDrive client. Everything was working fine until I went to the office and got another IP address which broke the ability of the OneDrive client to sign in.

6.4.1 So, what was going on?

The client defined a Conditional Access policy, which either requires a compliant device or a trusted location before access to SharePoint Online is granted. This caused my own OneDrive syn client to not be able to login and therefore was not usable until my customer changed something in their Conditional Access policy, or until I remove the sync the the SharePoint Online library hosted in the tenant of my customer, but I can only do that when I'm back in the home office where my IP is whitelisted.

6.4.2 Lessons learned

When creating a Conditional Access policy, this can have an unexpected impact even on Guest users. And in my opinion using IP addresses to exclude in a Conditional Access policy is asking for issues sooner or later. Go for Zero trust always.

6.5 Where to find help and provide feedback

There are many resources where you can find help on Azure Active Directory Conditional Access, when troubleshooting at first Google/Bing is your best friend here. If those search engines don't give the expected result, you can always ask at the following forums or reach out using twitter and other social media channels.

- Azure Active Directory @ MSDN - <https://social.msdn.microsoft.com/Forums/en-US/home?forum=WindowsAzureAD>
- Azure Active Directory @ Stack Overflow - <https://stackoverflow.com/questions/tagged/azure-active-directory>

If something does not work as expected another good source could be to check the Azure Active Directory user voice page, where a certain functionality might already be noticed by somebody else who requested the product team to solve it. You can find the UserVoice page for Conditional Access here: <https://feedback.azure.com/forums/169401-azure-active-directory/category/167259-conditional-access>





7 Modifying Conditional Access to suit your special needs.

When you want to integrate other products into your Conditional Access environment you can use "Custom controls" to include products from other vendors into your Conditional Access conditions. If a custom control is used the browser is redirected to the external service, performs any required authentication or validation activities, and is then redirected back to Azure Active Directory. If the user was successfully authenticated or validated, the user continues in the Conditional Access flow.

More information and some samples can be found here: [Custom controls \(preview\)](#)

Another thing you can do to extend the grant control with Terms of Use which users must consent with before they can access the cloud app. More information about creating the terms of use can be found here: Azure Active Directory terms of use - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/terms-of-use>

In the example below I have created the terms of use for my tenant Insight24

New terms of use

Terms of use

Create and upload documents

* Name 1

I24 Terms of Use



* Display name 1

Insight24 Terms of Use



Terms of use document 1

"I24 - Terms of use.pdf"



English



+ Add language

Require users to expand the terms of use 1

On Off

Require users to consent on every device 1

On Off

Expire consents 1

On Off

Duration before re-acceptance required (days) 1

90



Conditional access

* Enforce with conditional access policy templates

Create conditional access policy later



This terms of use will appear in the grant control list when creating a conditional access policy.



Create





Once created if I open any Conditional Access policy, I have an extra control available which I can select in the Grant control. In this case the user is granted access to the cloud app if the I24 Terms of Use are accepted by the user.

Grant

Select the controls to be enforced.

Block access
 Grant access

- Require multi-factor authentication ⓘ
- 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 (preview) ⓘ
See list of policy protected client apps
- I24 Terms of Use

For multiple controls

Require all the selected controls
 Require one of the selected controls

Select





8 Resources and further references

8.1 Microsoft documentation

- What is Conditional Access? - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/overview>
- What is Azure Active Directory? - <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-whatis>
- Azure Active Directory pricing - <https://azure.microsoft.com/en-us/pricing/details/active-directory/>
- Azure AD Adoption kits: <https://www.microsoft.com/en-us/download/details.aspx?id=58321>
- Microsoft 365 Business Service Description - <https://docs.microsoft.com/en-gb/office365/servicedescriptions/microsoft-365-business-service-description>
- QuickStart: Block access when a session risk is detected with Azure Active Directory conditional access - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/app-sign-in-risk>
- Infographic: Control access to your data with intelligence using Microsoft EMS - <https://gallery.technet.microsoft.com/Infographic-Control-access-81e7d79e>
- Infographic: Comprehensive protection of Office 365 data on any device with EMS - <https://gallery.technet.microsoft.com/Infographic-Comprehensive-e9a6c8c3>
- Enable combined security information registration (preview) - <https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-registration-mfa-sspr-combined>
- Enabling limited access with SharePoint Online - <https://aka.ms/spolimitedaccessdocs>
- Enabling limited access with Exchange Online - <https://aka.ms/owalimitedaccess>
- Use app enforced restrictions - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/concept-conditional-access-session#application-enforced-restrictions>
- Protect apps with Microsoft Cloud App Security Conditional Access App Control - <https://docs.microsoft.com/en-us/cloud-app-security/proxy-intro-aad#how-it-works>
- User sign-in frequency - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/howto-conditional-access-session-lifetime#user-sign-in-frequency>
- Configure authentication session management with Conditional Access - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/howto-conditional-access-session-lifetime>
- What are service dependencies in Azure Active Directory Conditional Access? - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/service-dependencies>
- Reduce your attack surface - <https://docs.microsoft.com/en-us/azure/security/azure-ad-secure-steps#step-2---reduce-your-attack-surface>
- Tutorial: Secure user sign-in events with Azure Multi-Factor Authentication - <https://docs.microsoft.com/en-us/azure/active-directory/authentication/tutorial-enable-azure-mfa>
- Quickstart: Block access when a session risk is detected with Azure Active Directory Conditional Access - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/app-sign-in-risk>
- Quickstart: Require terms of use to be accepted before accessing cloud apps - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/require-tou>
- Control access from unmanaged devices - <https://docs.microsoft.com/en-us/sharepoint/control-access-from-unmanaged-devices>





- Use sensitivity labels to protect content in Microsoft Teams, Microsoft 365 groups, and SharePoint sites - <https://docs.microsoft.com/en-us/microsoft-365/compliance/sensitivity-labels-teams-groups-sites?view=o365-worldwide#enable-this-preview-and-synchronize-labels>
- Assign sensitivity labels to Microsoft 365 groups in Azure Active Directory - <https://docs.microsoft.com/en-us/azure/active-directory/enterprise-users/groups-assign-sensitivity-labels>
- Use sensitivity labels to protect content in Microsoft Teams, Microsoft 365 groups, and SharePoint sites - <https://docs.microsoft.com/en-us/microsoft-365/compliance/sensitivity-labels-teams-groups-sites?view=o365-worldwide>
- Optimize reauthentication prompts and understand session lifetime for Azure Multi-Factor Authentication - <https://docs.microsoft.com/en-us/azure/active-directory/authentication/concepts-azure-multi-factor-authentication-prompts-session-lifetime>
- What is a Primary Refresh Token? - <https://docs.microsoft.com/en-us/azure/active-directory/devices/concept-primary-refresh-token>
- Configure the 'Stay signed in?' prompt for Azure AD accounts - <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/keep-me-signed-in>
- Configure authentication session management with Conditional Access - <https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/howto-conditional-access-session-lifetime>
- Azure AD registered devices - <https://docs.microsoft.com/en-us/azure/active-directory/devices/concept-azure-ad-register>
- Accessing Conditional Access protected resources in Microsoft Edge - <https://docs.microsoft.com/en-us/deployedge/security-overview#accessing-conditional-access-protected-resources-in-microsoft-edge>
- Enable passwordless sign-in with the Microsoft Authenticator app (preview) - <https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-authentication-passwordless-phone>
-

8.2 Other interesting blogs

- Conditional Access posts by Peter van der Woude - <https://www.petervanderwoude.nl/post/category/microsoft-intune/conditional-access/>
- Conditional Access posts by Peter Daalmans - <https://www.configmrblog.com/tag/conditional-access/>
- Conditional Access posts by Per Larsen - <https://osddeployment.dk/tag/conditional-access/>
- Conditional Access posts by Daniel Chronlund - <https://danielchronlund.com/>
- Conditional Access posts by Thomas Naunheim - <https://www.cloud-architekt.net/>
- How to get started with Conditional Access, by Per Larsen - <https://osddeployment.dk/2018/07/01/how-to-get-started-with-conditional-access/>
- Conditional Access - are you really getting the most out of it?, by Joni Nieminen - <https://bloggerz.cloud/2019/01/02/conditional-access-are-you-really-getting-the-most-out-of-it-part-2-of-2/>
- Implementing Modern Security Tools – Part 3 – Conditional Access, by Maurice Daly - <https://www.scconfigmgr.com/2019/02/19/implementing-modern-security-tools-part-3-conditional-access/>
- Azure Active Directory and Office 365: Conditional Access, by Jethro Seghers - <https://regarding365.com/azure-active-directory-and-office-365-conditional-access-8bc616a392b2>





- My favorite Conditional Access Policies for the SMB, by Alex Fields -
<https://www.itpromentor.com/conditional-access-faves/>
- Conditional access (zero trust) is the most important EUC movement since mobile and cloud, by Jack Madden - <https://www.brianmadden.com/opinion/Conditional-access-zero-trust-is-the-most-important-EUC-movement-since-mobile-and-cloud>
- Diverse articles on Conditional Access from the Practical 365 team -
<https://practical365.com/tag/conditional-access/>
- Bypassing Conditional Access Device Platform Policies, by Nicola Suter
- <https://tech.nicolonsky.ch/bypassing-conditional-access-device-platform-policies/>

