

Azure Conditional Access and Jamf Connect

Overview: This covers the following topics:

- Administrators finding "failed" login attempts in Azure sign-in logs when using Jamf Connect
- How the ROPG workflow and Jamf Connect communicate
- How to make an App Registration in Microsoft Azure Active Directory that allows for Conditional Access policies
- How to make Conditional Access policies NOT apply to the ROPG workflow

Problem:

Administrators may observe failed login attempts in the log for the enterprise application created in Microsoft Azure Active Directory when using Jamf Connect and a Conditional Access policy that requires multi-factor authentication for the target of "All cloud apps." While this is expected behavior of the Resource Owner Password Grant (ROPG) workflow, it may trigger a user appearing in the Risky Sign-Ins in Azure Active Directory security reports.

What is happening:

The target of "All cloud apps" applies policies far beyond the logins to specific cloud services and applies policies to non-interactive workflows like those with ROPG.

Specifically, the "All cloud apps" appears to apply to any application requesting a login with the scope of any of the following: `openid profile email`

The Open ID Connect 2.0 specification uses these default scopes to obtain an access or identity token for a client application. Consequently, in its default configuration, Jamf Connect login uses the `openid profile email` scope, and the only way to apply a CA policy in this default behavior is to apply the policy to "All cloud apps" with NO exceptions applied or the CA policy will break.

Administrators have multiple options for solutions to enforce MFA on the Jamf Connect login screen:

- **Simplest, but most impact on user logins:** Set hard requirements for MFA via the older method of Azure Multi-Factor Authentication which applies an MFA requirement to ALL logins to ANY service for a specific user. Ignore failed logins in the sign-in logs for ROPG checks of the password. (Additional information on how to determine if a failed login is due to Jamf Connect menu bar agent doing an ROPG request is below.)
- **Simple, but may affect other services:** Apply a Conditional Access policy applied to "All cloud apps" requiring Multi-factor Authentication for login. Do NOT use an exception to the policy as that appears to break functionality of the CA rule as of testing done 10DEC2021. Ignore failed logins in the sign-in logs for ROPG checks of the password.
- **Complex, but exacting:** Follow the instructions in the post "Creating a custom scope for Jamf Connect in Azure for Conditional Access policies" to create a custom scope for Jamf Connect applications. Verify that no policies are created that apply to "All cloud apps" as to not affect the ROPG workflow. CA policy will be applied as expected to the Jamf Connect login application and ROPG check will appear as a successful login in sign-in logs.

Azure Multi-factor Authentication vs. Conditional Access

Administrators can enable multi-factor authentication requirements for a user account in two ways:

- Multi-factor Authentication which is reachable via the "All services" list in the Azure portal
- Conditional Access which is reachable via Azure Active Directory under Security

Multi-factor Authentication is a system wide, all login attempts, master switch system for enforcing MFA at authentication. While IP address ranges can be exempted, the rules apply to all authentications.

Conditional Access allows for fine grain details to apply for when MFA is required including exempting MFA for web applications.

Resource Owner Password Grant workflow

Jamf Connect uses a Resource Owner Password Grant (ROPG) workflow to synchronize the user's password in the identity provider with the password on the user's client machine. The user name and the password are sent to the identity provider in a "non-interactive" login to receive a response. This means that the user is

not prompted for any sort of user name or password when logging in; Jamf Connect is using the information securely stored in the user's keychain for this event.

For Azure, the responses are one of the following:

- **Success, no MFA requirements:** An access, refresh, and ID token encoded in HS256
- **Success, MFA required through a policy:** An error response like `AADSTS50076:`
Due to a configuration change made by your administrator, or because you moved to a new location, you must use multi-factor authentication to access [application UUID]
- **Failure, bad password or user name:** An error response like `AADSTS50126:`
Error validating credentials due to invalid username or password.

As long as the user password is correct, the ROPG flow has succeeded - the password has been validated to be correct. Whereas Jamf Connect has no need for the access, refresh, and ID token to keep the local password in sync with the identity provider, an appropriate error response is interpreted as a successful password check.

Reference: https://docs.jamf.com/jamf-connect/administrator-guide/Authentication_Protocols.html

Diagnosing MFA vs. failed password in Azure logs

Navigate to Azure Active Directory → Enterprise Applications and select the name of your Jamf Connect application in Azure. Navigate to Activity → Sign-ins to open user usage logs.

Date	Request ID	User	Application	Status	IP address	Location	Conditional access	Authentication requi...
3/2/2021, 1:37:02 PM	d8fe2ab1-5896-401a-8c9...	Sean Rabbitt	Jamf Connect - jamfse.io ...	Failure	206.214.236.79	San Jose, California, US	Not Applied	Multi-factor authentication
3/2/2021, 1:36:55 PM	7bd1116-89ab-4ca8-8a5...	Sean Rabbitt	Jamf Connect - jamfse.io ...	Failure	206.214.236.79	San Jose, California, US	Not Applied	Single-factor authenticati...

Example logs from a sample Microsoft Azure instance

Shown above are two logins which appear to be failures. Under the "Authentication required" column, the first login says "Multi-factor authentication". Clicking on the row will pull up additional details about the login attempt.

Details						
Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only	Additional Details
Authentication Policies Applied						
Per-user MFA						
Date	Authentication method	Authentication method...	Succeeded	Result detail	Requirement	
3/2/2021, 1:37:02 PM	Password	Password Hash Sync	false	User did not pass the MF...	User	
3/2/2021, 1:37:02 PM			false	MFA required in Azure AD	User	

Details on a non-interactive login from Jamf Connect via ROPG

Under Authentication Details, the “Result detail” will let an administrator determine if the login was successful or a failure. In this example, the login was a success - the Result detail shows that the “User did not pass the MFA challenge (non interactive).” This login can be interpreted in that the user was required to use MFA by either a Conditional Access policy or through Azure Multi-factor authentication. In the second example, a user with MFA required failed to enter their correct password:

Details						⌵	
Basic info	Location	Device info	Authentication Details	Conditional Access	Report-only	Additional Details	⋮
Date	Authentication met...		Authentication met...	Succeeded	Result detail		
3/2/2021, 1:36:55 PM	Password		Password Hash Sync	false	Invalid username or password or Invalid on-premise username or passwo		

Details on a failed non-interactive login due to an incorrect password

The Authentication required column shows “Single-factor authentication” and the Authentication Details show “Invalid username or password or Invalid on-premise username or password.” While the user is required to use Multi-factor authentication, the user failed the first, single factor and thus was never prompted for MFA.

Creating a Custom Scope for Jamf Connect and Conditional Access policies

Workflow overview:

- Create a “private endpoint” application registration with a custom API
 - With API permissions for “User.read”
 - With “Expose an API” scope created
- Create a “public endpoint” application registration for OIDC to call that custom API

- Remove API permissions for “User.read”
- Add API permission for “My APIs” for the name of the application created in first step and the scope created in first step
- Create an Azure Conditional Access policy to require multifactor authentication
 - Apply to application created in first step
- Remove any CA policy applied to “All cloud apps” that would require MFA
- Create an application registration for ROPG
 - Follow standard instructions for creating an application for Jamf Connect
 - Optional: Remove API permission for User.read completely - no API permissions needed
- Create a Jamf Connect Login configuration profile
 - Custom IDP
 - Use Discovery URL set to the V2 endpoints for Azure - [https://login.microsoftonline.com/\[tenant ID\]/v2.0/.well-known/openid-configuration](https://login.microsoftonline.com/[tenant ID]/v2.0/.well-known/openid-configuration)
 - Set OIDCScopes to the scope created in the first step
 - Set OIDCClientID to the SECOND application

Step One: Create an application registration with a custom API

Navigate to portal.azure.com → Azure Active Directory → App Registrations. Create a new app registration. Name the application something like “Jamf Connect - Conditional Access Policy API”. Select the supported account types to “Accounts in this organizational directory only”. Leave Redirect URI section blank. Register the application.

Home > jamfse.io >

Register an application ...

* Name

The user-facing display name for this application (this can be changed later).

Jamf Connect - Conditional Access Policy API

✓

Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory only (jamfse.io only - Single tenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- ☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web

e.g. https://example.com/auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the [Microsoft Platform Policies](#) 

Register

Application Registration Screen (as of 06DEC2021)

Navigate to API permissions on the left hand navigation bar. Grant admin consent for the organization.

Jamf Connect - Conditional Access Policy API | API permissions

Search (Cmd+ /)

Refresh

Got feedback?

Overview

Quickstart

Integration assistant

Manage

Branding

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators | Preview

Manifest

Support + Troubleshooting

Troubleshooting

New support request

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect th

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission

☒ Grant admin consent for

API / Permissions name	Type	Description	Admin consent requ...	Status
▼ Microsoft Graph (1)				
User.Read	Delegated	Sign in and read user profile	No	...

To view and manage permissions and user consent, try [Enterprise applications](#).

Using the left hand navigation bar, select “Expose an API”. Set the Application ID URI. A default entry will be created based on the pattern of `api://[application ID]`. This may be modified if desired but default entry is acceptable.

Jamf Connect - Conditional Access Policy API | Expose an API

Search (Cmd+ /)

Got feedback?

Overview

Quickstart

Integration assistant

Manage

Branding

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators | Preview

Manifest

Support + Troubleshooting

Troubleshooting

New support request

Set the App ID URI

Application ID URI

api://66b2a554-0863-44be-a8e6-303cd3645b3c

Save

Discard

Adding a scope here creates only delegated permissions. If you are looking to create application-only scopes, use 'App roles' and define app roles assignable to application type. [Go to App roles](#).

+ Add a scope

Scopes	Who can consent	Admin consent display ...	User consent display na...	State
No scopes have been defined				


Authorized client applications

Authorizing a client application indicates that this API trusts the application and users should not be asked to consent when the client calls this API.

+ Add a client application

Client Id	Scopes
No client applications have been authorized	

Select the option for “Add a scope”

 Got feedback?

Application ID URI

Scopes defined by this API

Define custom scopes to restrict access to data and functionality protected by the API. An application that requires access to parts of this API can request that a user or admin consent to one or more of these.

Adding a scope here creates only delegated permissions. If you are looking to create application-only scopes, use 'App roles' and define app roles assignable to application type. [Go to App roles](#).

[+ Add a scope](#)

Scopes	Who can consent	Admin consent display ...	User consent display na...	State
No scopes have been defined				


Authorized client applications

Authorizing a client application indicates that this API trusts the application and users should not be asked to consent when the client calls this API.


[+ Add a client application](#)


Client Id	Scopes
No client applications have been authorized	


Add a scope


Scope name * 


api//66b2a554-0863-44be-a8e6-303cd3645b3c/


Who can consent? 
☐ Admins and users ☒ Admins only

Admin consent display name * 

Admin consent description * 

User consent display name 

User consent description 

State 
☒ Enabled ☐ Disabled

[Add scope](#) [Cancel](#)

Set the scope name to `jamfconnect`. Set "Who can consent" to "Admins". Enter information into the Admin consent display name and Admin consent description.

Any text is acceptable - this will be accepted by the admin in the next step. Press "Add scope" to save.

Add a scope



Scope name * ⓘ

jamfconnect



api://66b2a554-0863-44be-a8e6-303cd3645b3c/jamfconnect

Who can consent? ⓘ

Admins and users Admins only

Admin consent display name * ⓘ

Read user information



Admin consent description * ⓘ

Allows Jamf Connect to read user information like user name, email address, real name, role, and group membership if required.



User consent display name ⓘ

Read user information



User consent description ⓘ

Users should never see this description unless an administrator has failed to grant consent for the organization.

State ⓘ

Enabled Disabled

Add scope

Cancel

Copy the scope with the Copy button and save it for later. This will be used as the `OIDCScopes` later in Jamf Connect Configuration.

Step Two: Create an application registration using this new API permission

Return to Azure Active Directory → App Registrations. Create a new app registration. Name the app “Jamf Connect - OIDC Endpoint”. Set Supported account types to “Accounts in this organizational directory only”. Set Redirect URI to “Public client/native (mobile & desktop)” with the value `https://127.0.0.1/jamfconnect`. Register the application.

Register an application ...

* Name

The user-facing display name for this application (this can be changed later).

Jamf Connect - OIDC Endpoint



Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory only (jamfse.io only - Single tenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- ☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Public client/native (mobile ...



https://127.0.0.1/jamfconnect



Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the [Microsoft Platform Policies](#)

Register

Navigate to API permissions. By default, the Microsoft Graph → User.Read permission is added. Use the ellipses to the right of Status to remove this permission from the application. Next, select "+ Add a permission".

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission ✓ Grant admin consent for jamfse.io

API / Permissions name	Type	Description	Admin consent requ...	Status	
Microsoft Graph (1)					
User.Read	Delegated	Sign in and read user profile	No	Granted for jamfse.io	Remove permission Revoke admin consent

To view and manage permissions and user consent, try [Enterprise applications](#).

Select the “My APIs” tab. Select the name of the application you created in step 1.

Request API permissions

Select an API

Microsoft APIs APIs my organization uses My APIs

Applications that expose permissions are shown below


Name	Application (client) ID
Jamf Setup - Retail	3c272147- 
Jamf Connect - Conditional Access Policy API	66b2a554- 
Jamf Connect - INFOSEC ONLY ACCESS	b92961e0- 

Select the option for “Delegated permissions” and check the box for “jamfconnect” - the only permission listed in the application. Use the “Add permissions” button to close the window.

Request API permissions

×

< All APIs

 Jamf Connect - Conditional Access Policy API
api://66b2a554-0863-44be-a8e6-303cd3645b3c

What type of permissions does your application require?


Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions

expand all

 Start typing a permission to filter these results

 The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

×

Permission	Admin consent required
Permissions (1)	
<input checked="" type="checkbox"/> jamfconnect ⓘ Read user information	No

Add permissions

Discard

Use the “Grant admin consent for [domain]” to grant permission to access the API on behalf of users.

Navigate to App roles and create an app role for Admin and Standard to correspond to users who will be granted administrator rights on macOS client devices.

Navigate to Overview. Record the Application (client) ID and the Directory (tenant) ID for later use with Jamf Connect Configuration.

^ Essentials

Display name : [Jamf Connect - OIDC Conditional Access](#)

Application (client) ID : baf44d07-

Object ID : c520d014-

Directory (tenant) ID : f83fb0da-

Supported account types : [My organization only](#)

Navigate to Azure Active Directory → Enterprise Applications. Find the Jamf Connect - OIDC Endpoint application you created and assign users and roles to the application.

Step Three: Create an Azure Conditional Access policy

Navigate to portal.azure.com → Azure Conditional Access. Create a new policy.

[Home](#) > [Conditional Access](#)



Conditional Access | Policies

Azure Active Directory



+ New policy ▾



What If



Refresh



Overview (Preview)



Policies



Insights and reporting



Diagnose and solve problems

Create new policy

Create new policy from templates (Preview)



To improve the resilience of Azure AD, we are an

Name the policy as desired. The sample will name the policy "Jamf Connect - Require Multifactor Authentication"

New ...

Conditional Access policy

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

Name *

Jamf Connect - Require Multifactor Auth...✓

Assignments

Users or workload identities ⓘ

0 users or workload identities selected

Cloud apps or actions ⓘ

No cloud apps, actions, or authentication contexts selected

Conditions ⓘ

0 conditions selected

Access controls

Grant ⓘ


0 controls selected

Session ⓘ

0 controls selected

Enable policy

Report-onlyOnOff

 Do not block yourself out! This policy impacts the Azure portal and other clients that do not support CAE today.

Create

Select “Users or workload identities”. Select a test user to test your conditional access policy before applying to all users.

[Home](#) > [Conditional Access](#) >

New ...

Conditional Access policy

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies.

[Learn more](#)

Name *

Jamf Connect - Require Multifactor Auth... ✓

Assignments

Users or workload identities ⓘ

[Specific users included](#)

✗ "Select users and groups" must be configured

Cloud apps or actions ⓘ

[No cloud apps, actions, or authentication contexts selected](#)

Conditions ⓘ

[0 conditions selected](#)

Access controls

Grant ⓘ

Control access based on who the policy will apply to, such as users and groups, workload identities, directory roles, or external guests.

[Learn more](#)

What does this policy apply to?

Users and groups ▼

Include Exclude

☐ None

☐ All users

☒ Select users and groups

☐ All guest and external users ⓘ

☐ Directory roles ⓘ

☒ Users and groups

Select

[0 users and groups selected](#)

✗ Select at least one user or group

Select “Cloud apps or actions”. Select the Jamf Connect - Conditional Access Policy API you created in step one.

[Home](#) > [Conditional Access](#) >

New

Conditional Access policy

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

Name *

Jamf Connect - Require Multifactor Auth... ✓

Assignments

Users or workload identities ⓘ

[Specific users included](#)

Cloud apps or actions ⓘ

No cloud apps, actions, or authentication contexts selected

✗ "Select apps" must be configured

Conditions ⓘ

Control 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




Select

[None](#)

✗ Select at least one app.

Select

Cloud apps

- ☐  Jamf Connect - Conditional Access Policy API
- ☐  [Redacted]
- ☐  [Redacted]

Selected items

Select "Grant". Check the option for "Require multi-factor authentication". Set Enable policy to "On" and "Create" to save the policy.

New ...

Conditional Access policy

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

Name *

Jamf Connect - Require Multifactor Auth... ✓

Assignments

Users or workload identities ⓘ

[Specific users included](#)

Cloud apps or actions ⓘ

[1 app included](#)

Conditions ⓘ

[0 conditions selected](#)

Access controls

Grant ⓘ

[0 controls selected](#)

Session ⓘ

[0 controls selected](#)

Enable policy

Report-only On Off

⚠ Do not block yourself out! This policy impacts the Azu today.

Create

Grant



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

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

☐ Require password change ⓘ

☐ RequireDuoMfa

For multiple controls

☒ Require all the selected controls

☐ Require one of the selected controls

Select

Step Four: Remove any Conditional Access policies applied to All cloud apps

Navigate to portal.azure.com → Azure Conditional Access. Examine any application applied to the scope of “All cloud apps”. Either set “Enable policy” to “Off” for any application that has a Grant of “Require multi-factor authentication” or modify the “Cloud apps or actions” to specifically list resources that should have MFA applied.

Applying a policy to require MFA for “All cloud apps” will cause the ROPG application in the next step to inaccurately show failed logins in the Azure sign-in logs.

Step Five: Create an application registration for ROPG

Follow the instructions in https://docs.jamf.com/jamf-connect/documentation/Integrating_with_Microsoft_Azure_AD.html to create an application for ROPG.

OPTIONAL: The ROPG application does not need any API permissions. Navigate to API permissions and revoke consent for the User.Read permission and delete the User.Read permission completely from the ROPG app. The access token granted during the ROPG check will be unable to access any resources.

Step Six: Create a Jamf Connect Configuration Profile

Create settings with the Custom identity provider. Set the OIDC Client ID to the application ID created in step two. Set the ROPG client ID to the application ID created in step five. Set the Scopes to value you saved as the last part of step one.

Set OIDC Redirect URI to <https://127.0.0.1/jamfconnect>. Set the discovery URL to [https://login.microsoftonline.com/\[ORGANIZATION_TENANT_ID\]/v2.0/.well-known/openid-configuration](https://login.microsoftonline.com/[ORGANIZATION_TENANT_ID]/v2.0/.well-known/openid-configuration) where the tenant ID is the UUID of your Azure tenant.

Identity Provider
Login
Connect

Required

Identity Provider: Custom

OIDC Client ID: 2520beb2-

Advanced OIDC

Scopes: api://66b2a554-0863-44be-a8e6-303cd3645b3c/jamfconnect

Token Caching: ☐ Ignore cookies

ROPG Client ID: 9fcc52c7-ee36-4889-8517-lkjslkjoe23

Client Secret: JCCyfVL7YWtP6gudLljRZV_N0dW4f3xEtlxqtokEAZ6FAsBtgylq0MpU1uQ7Jid

Tenant: login.myidp.com/c27d1b33-59b3-4ab2-a5c9-23jf0093

OIDC Redirect URI: https://127.0.0.1/jamfconnect

Discovery URL: ftonline.com/f83fb0da- /v2.0/.well-known/openid-configuration

Choose License... Jamf Connect operates in trial mode without a license

Test your configuration with the test user via OIDC. Make sure MFA was required.

To test the ROPG step, make a second configuration in Jamf Connect Configuration.

Copy the configuration from the first configuration. REMOVE the OIDCScopes from the configuration. Use the Test → ROPG button to simulate what happens in the production software when ROPG checks are sent with the scope `openid profile email`. Check response to make sure that ROPG did not prompt for an MFA response to get an identity or access token.