

# Experiences with Multi Admin Approval

Multi Admin Approval is a feature in Intune, that require a second administrative account to approve a change before the change is applied.

With Multi Admin Approval (MAA), you configure access policies that protect specific configurations, like Apps or Scripts for devices. Access policies specify what is protected and which group of accounts are permitted to approve changes to those resources.

When any account in the Tenant is used to make a change to a resource that's protected by an access policy, Intune won't apply the change until a different account explicitly approves it. Only administrators who are members of an approval group that's assigned a protected resource in an access protection policy can approve changes. Approvers can also reject change requests.

## Field report

- This feature is currently only applicable for Intune apps and Windows / MacOS scripts.
- To create or approve an approval request the account needs the role Intune Administrator even when in the account is in the approver group.
- The appropriately protected Intune resources (apps, scripts) cannot be restricted individually but are tenant wide protected for everyone via Multi Admin Approval.
- At the time of writing every request with scripts is only valid for one hour and then the status changes to expired. This does not apply to the Intune applications.
- Following entity actions need a separate approval request, whenever one of the actions is performed:
  - Edit
  - Create
  - Modify
  - Delete
  - Assign

## Steps of approval requests

After doing a described action (create, modify, delete, etc.) on an Intune resource which is protected by an access policy, will create an approval request in the Intune Admin Center. To submit the change you can use the normal Intune Admin Center.

Summary

⚠ Before this resource can be created, it must be approved by another admin. Before you can submit this request, you must enter your business justification.

Basics

Name PSS-INT-LNC-PS1-TestMultiAdminApproval-NONPROD  
Description --

Script settings

PowerShell script Create-ConfirmationBoxes.ps1  
Run this script using the logged on credentials No  
Enforce script signature check No  
Run script in 64 bit PowerShell Host No

Scope tags

Default

Business justification \*

Testing this feature with not productive workload

Previous    Submit for approval

# Needs approval

After that submission a new approval request is created in the Intune Admin Center which needs to be approved or rejected from an other administrator account.

Home > Tenant admin  
Tenant admin | Multi Admin Approval ...

Search

Tenant status  
Remote help  
Microsoft Tunnel Gateway  
Connectors and tokens  
Filters  
Roles  
Azure AD Privileged Identity Management  
Diagnostics settings  
Audit logs  
Device diagnostics  
Multi Admin Approval  
Premium add-ons  
End user experiences  
Customization  
Organizational messages (preview)  
Custom notifications  
Terms and conditions  
Windows Autopatch  
Tenant enrollment  
Microsoft Managed Desktop

Received requests    My requests    Access policies

Refresh Columns

Search by justification

Add filter

Showing 1 to 1 of 1 records

Requested on	Resource type	Operation	Business justification
12.12.2022, 12:03:24	Powershell script	Create	Testing this feature with not productive workload.

Testing this feature with not productive workload.  
Multi admin approval request

Review the changes below and take the appropriate action.

Request information

Resource type	Powershell script
Operation	Create
Requested by	
Requested on	12.12.2022, 12:03:24
Business justification	Testing this feature with not productive workload.

Property	Requested changes
enforceSignatureCheck	No
runAs32Bit	Yes
displayName	PSS-INT-LNC-PS1-TestMultiAdminApproval-NONPROD
runAsAccount	
fileName	Create-ConfirmationBoxes.ps1
roleScopeTagIds	Default

Script content changes

```
1 --
2 $i=1
3 while($i -lt 6){
4     Add-Type -AssemblyName PresentationCore,PresentationFramework
5     $ButtonType = [System.Windows.MessageBoxButton]::YesNoCancel
6     $MessageIcon = [System.Windows.MessageBoxImage]::Error
7     $MessageBody = "Do you want to download a virus?"
8     $MessageTitle = "Confirm youve been hacked"
9     $Silverselection = [System.Windows.MessageBox]::Show($MessageBody,$MessageTitle,$ButtonType,$MessageIcon)
10 }
11 }
```

This approval is then moved to approved or rejected status according to the selection made.

# Rejected

Rejected means that no further actions have to be made. The entity is archived and the status is set to rejected.

# Approved

When approved by a different administrator your entity is then forwarded back to you so you can deploy the change at a time when it suits the creating person. The apps are implemented directly, without this following steps. This was tested with the scripts.

Approver notes

test

Complete request

When "Complete request" is pressed by the owner of the approval request, the deployment of the change starts and gets implemented accordingly. The request then changes to the state of "Completed".

## Completed

Completed are all requests which where approved by a different administrator and deployed by the owner. These changes were effectively made to the environment.

The Multi Administrator Approval is also very practical to trace changes.

Received requests My requests Access policies

Refresh Columns

Search by justification Add filter

Showing 1 to 9 of 9 records

Requested on	Resource type	Operation	Business justification	Requested by	Status
12.12.2022, 13:43:59	Powershell script	Create	13:43		Expired
12.12.2022, 13:42:08	Powershell script	Delete	delete this as well		Completed
12.12.2022, 13:41:54	Powershell script	Delete	please delet dis		Completed
12.12.2022, 13:38:47	Powershell script	Create	please		Completed
12.12.2022, 13:38:26	Powershell script	Delete	test right click		Completed
12.12.2022, 13:37:12	Powershell script	Assign	deploy to group		Rejected
12.12.2022, 13:27:49	Powershell script	Create	weil gebraucht.		Completed
12.12.2022, 12:09:35	Powershell script	Create	test2 not productive.		Expired
12.12.2022, 12:03:24	Powershell script	Create	Testing this feature with not productive workload.		Expired

## Expired

All requests which are not applied in one hour will get the status "Expired".

# Create access policy

To create an access policy, you can change to "Multi Admin Approval" under "Tenant administration". There under "Access policies" you can create a new policy.

Tenant admin | Multi Admin Approval

Search

Tenant status

Remote help

Microsoft Tunnel Gateway

Connectors and tokens

Filters

Roles

Azure AD Privileged Identity Management

Diagnostics settings

Audit logs

Device diagnostics

Multi Admin Approval

Alerts (preview)

Premium add-ons

Received requests

My requests

Access policies

Access policies allow you to control which tasks and actions need approval along with the specific approval groups

Create

Refresh

Search by name

Showing 1 to 2 of 2 records

Name

MAAAP-INT-LNC-PS1-AllAdminApprovalApps-NONPROD

tst-caprl

First you have to name the policy and choose the Profile type. Currently there are only two options; Scripts and Apps to select.

### Create an access policy

1 Basics

2 Approvers

3 Review + create

Name \*

MAAAP-INT-LNC-PS1-AllAdminApprovalApps-NONPROD

Description

Profile type \* ⓘ

Scripts

ⓘ

A script policy will limit any action on a script. These actions include create, edit, assign, and delete.

In addition, the approver group must be selected there. This group must contain the accounts which are authorized to approve or reject approval requests. These accounts must have to activate the "Intune Administrator" role.

✓ Basics

✓ Approvers

3 Review + create

Summary

Basics

Name

MAAAP-INT-LNC-PS1-AllAdminApprovalApps-NONPROD

Description

--

Profile type

scripts

Approvers

Included groups

AAD\_TEST\_CAPRL

Revision #8

Created 9 December 2022 14:58:10

Updated 21 July 2024 15:11:16