**Gettings started with Microsoft Teams Rooms (Part 1)**

by [Jeroen Burgerhout](https://burgerhout.org/author/jeroen/) 6 months ago  4 MIN READ

Before we are going to proceed with this series of blog posts, first question probably pops ups

What is a Microsoft Teams Room?

A **Microsoft Teams Room** is a device with special software installed which is the Teams Room software. This software is installed on top of a Windows 10 Enterprise or Windows 10 Enterprise IoT installation.

You can transform meeting spaces with these kind of devices to a rich and collaborative Teams experiences. For example: Instead of sitting with a few people in a meeting room where they all logged into Teams and have their microphone and speakers open, you can choose to install a Microsoft Teams Room device. You connect this device to a big size TV and connect a webcam to it.

You can manage the MTR’s with Endpoint Management and you can customize the MTR as well. But…. What if you want to play with it and cannot afford to buy a device from HP, Yealink, Logitech or other partners? Or you do not have a spare Surface Pro somewhere in the cupboard.

**You can install an MTR in Hyper-V!**

In this series of blog post, I want to help to create a Teams Room account, setup a virtual MTR, connect it Endpoint Management and push some configuration policies before you are pushing this to production MTR’s. And the last topic will be on how to customize the MTR. This series is divided in the following posts:

This series is divided in the following posts:  
Part 1: [Creating the Teams Room account and arrange the license](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-1/)  
Part 2: [Installing Microsoft Teams Room in Hyper-V](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-2/)  
Part 3: [Connect and manage it with Microsoft Endpoint Manager](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-3)  
Part 4: [Customize the Microsoft Teams Room](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-4/)  
Part 5: ?

You can find the files, scripts and other stuff that I have used in this series on my GitHub page at <https://github.com/BurgerhoutJ/scripts/tree/main/microsoft-teams-rooms>.

**Part 1: Creating the Teams Room account and arrange the license**

**Arranging the license**

First things first. Licensing! The key aspect to get a working MTR. You can use an expensive M365 E5 license for your MTR, but Microsoft is offering a special license for this, namely the **Meeting Room SKU**. In this license you have the following “apps”:

* Skype for Business
* Microsoft Teams
* Phone System
* Audio conferencing
* Microsoft Intune

I personally have a M365 developer tenant, so I created my own Meeting Room SKU with the above mentioned “apps”, except for the Audio conferencing, because this option is not available in your M365 developer tenant.

I created a security group in Azure AD with a dynamic membership rule, to populate the security group with the MTR-accounts, as shown as in Figure 1-1 and added the customized Meeting Room SKU to this security group as you can see in Figure 1-2.

Graphical user interface, text, application, email

Description automatically generatedFigure 1-1 : The dynamic membership rule for the security groupGraphical user interface, application

Description automatically generatedFigure 1-2 : The assigned license for the security group

The script I have created, is as like this:

# Connect to Exchange Online and create a resource

Set-ExecutionPolicy RemoteSigned

$UserCredential = Get-Credential

$Session = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionUri https://outlook.office365.com/powershell-liveid/ -Credential $UserCredential -Authentication Basic -AllowRedirection

Import-PSSession $Session -DisableNameChecking

New-Mailbox -Name "MTR-ConferenceRoom02" -Alias ConferenceRoom02 -Room -EnableRoomMailboxAccount $true -MicrosoftOnlineServicesID MTR-ConferenceRoom02@theorange.cat -RoomMailboxPassword (ConvertTo-SecureString -String '<Password>' -AsPlainText -Force)

Set-CalendarProcessing -Identity "MTR-ConferenceRoom02" -AutomateProcessing AutoAccept -AddOrganizerToSubject $false -DeleteComments $false -DeleteSubject $false -RemovePrivateProperty $false -AddAdditionalResponse $true -AdditionalResponse "This is a Microsoft Teams Room!"

#Sleep for 20 seconds

Start-Sleep 20

#Connect to MS Online to set password to never expires

Connect-MsolService -Credential $UserCredential

Set-MsolUser -UserPrincipalName MTR-ConferenceRoom02@theorange.cat -PasswordNeverExpires $true

Get-MsolUser -UserPrincipalName MTR-ConferenceRoom02@theorange.cat | Select PasswordNeverExpires

#Sleep for 300 seconds to populate the security group and assign a SfB account

Start-Sleep 300

#Enable account for SfB

Import-Module SkypeOnlineConnector

$cssess=New-CsOnlineSession -Credential $UserCredential

Import-PSSession $cssess -AllowClobber

$rm="MTR-ConferenceRoom02@theorange.cat"

Enable-CsMeetingRoom -Identity $rm -RegistrarPool "[sippoolDB42E12.infra.lync.com](http://sippooldb42e12.infra.lync.com/)" -SipAddressType EmailAddress

Copy

The script above, is named “CreatingTeamsRoomAccount.ps1” and you can find this in the GitHub repository.  
After you have created the MTR-account, you must give Azure AD and Skype for Business a few minutes to evaluate your security group, in order to assign your Meeting Room SKU license.

Let us see this script and the result in action in the recording below.

Watching the script in actionGraphical user interface, application

Description automatically generatedFigure 1-3 : The MTR user is created and is assigned to your Meeting Room SKUGraphical user interface, text, application, email

Description automatically generatedFigure 1-4 : The MTR resource account is created

That's it for this moment. See you soon for part 2 of this blog post series.

**Resources**

* <https://github.com/BurgerhoutJ/MicrosoftTeamsRoom>
* <https://docs.microsoft.com/en-us/microsoftteams/rooms/rooms-licensing>
* <https://developer.microsoft.com/en-us/microsoft-365/>
* <https://docs.microsoft.com/en-us/microsoftteams/rooms/rooms-configure-accounts>

**Gettings started with Microsoft Teams Rooms (Part 2)**

by [Jeroen Burgerhout](https://burgerhout.org/author/jeroen/) 6 months ago  4 MIN READ

Welcome at the second part of this series about the virtual Microsoft Teams Room. I had some busy weeks and I know that a few people are waiting for this part, so we are going to start.

In part 1, we have configured the MTR-account in Office365, assigned a Meeting Room license, created a resource mailbox with some optional parameters for automatic reply and so on, enabled Skype for Business (as a room) and set the password no not expire.

This series is divided in the following posts:  
Part 1: [Creating the Teams Room account and arrange the license](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-1/)  
Part 2: [Installing Microsoft Teams Room in Hyper-V](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-2/)  
Part 3: [Connect and manage it with Microsoft Endpoint Manager](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-3)  
Part 4: [Customize the Microsoft Teams Room](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-4/)  
Part 5: ?

You can find the files, scripts and other stuff that I have used in this series on my GitHub page at <https://github.com/BurgerhoutJ/scripts/tree/main/microsoft-teams-rooms>.

**Part 2: Installing Microsoft Teams Room in Hyper-V**

In this part we are going to install a Windows 10 device in Hyper-V, install the SRS software on it, tweak it a little with SkypeSettings.xml and the system will be ready for use.

**What do we need?**

You must have the following:

* Hyper-V (or something else)
* Windows 10 1909 Enterprise ISO
* Teams Rooms software
* Internet (of course)

At the time of writing this part, Windows 10 1909 (18363.418) is now supported with version 4.5.33.0 of the Teams Room software. Blog post is updated.

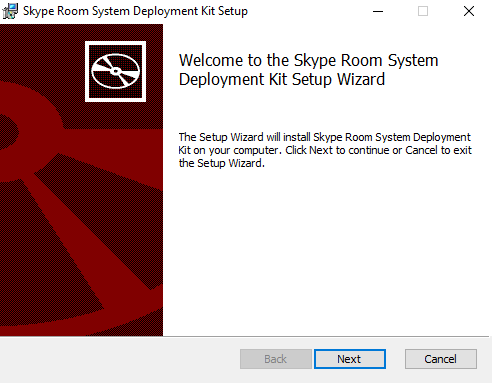
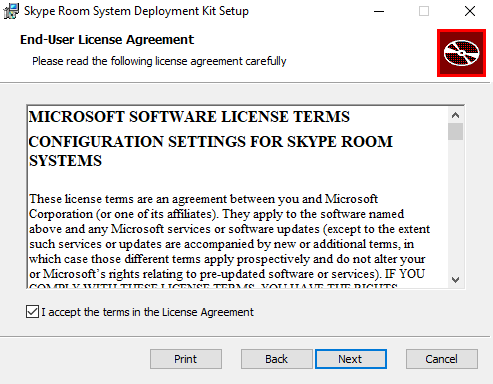
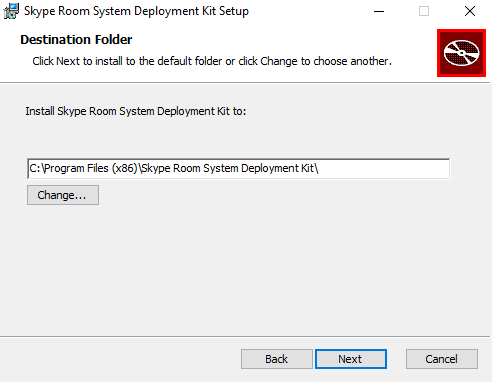
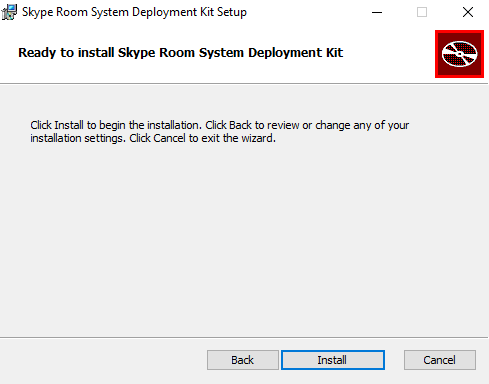
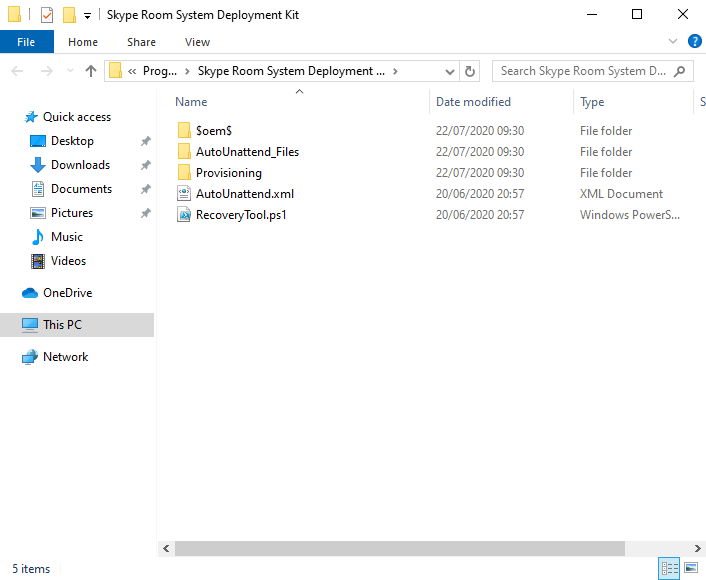
**Hyper-V**

In Hyper-V, we are going to create a new virtual machine with 8GB of RAM and mount the 1909 ISO to it. Boot it, proceed with the install wizard, create a local account, do some updates **(do not install 2004 or 20H2)** and done. That is quite straight forward.

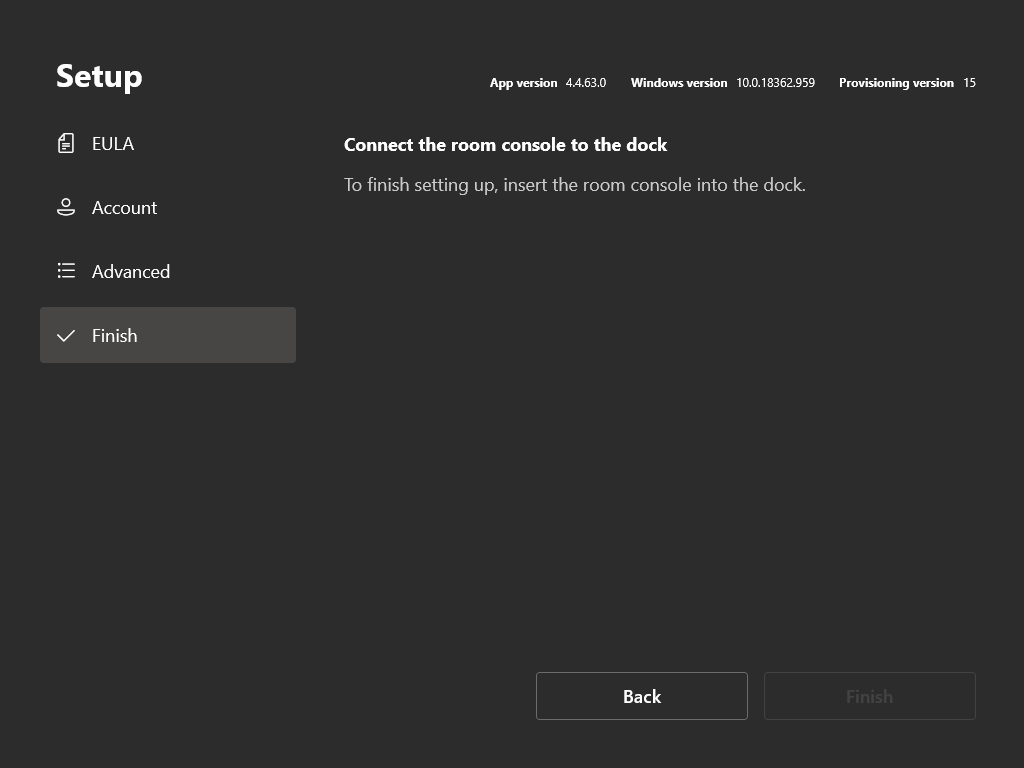
**Teams Rooms software**

You can download the MTR software for free at <https://go.microsoft.com/fwlink/?linkid=851168>. This morning I downloaded version 4.5.33.0 and this is also the version we are going to install.

After your virtual Windows 10 1909 device is ready, copy / paste the downloaded SRS software to a folder on the virtual machine. In my case I used C:\Temp.

1. **Execute** the *SRSDeploymentKit-4.5.33.0.msi*
2. **Accept** the License Agreement
3. **Choose** the desired folder to install
4. Click on **Install** to continue
5. A new explorer window will pop up
6. **Start** PowerShell in admin mode
7. Go to the folder where you installed the deployment kit
8. **Execute** Set-ExecutionPolicy Bypass
9. **Execute** MD C:\Recovery\OEM
10. **Execute** RecoveryTool.ps1
11. **Choose** for option **2**
12. **Open** the Settings app, Update & Security, Recovery and click on Get started under the Reset this PC option. Choose for Remove Everything, proceed with the wizard until the device will reset itself.

Lean back and grab a coffee and watch how this device magically convert to a Teams Room device. I will not be going to bore you with 45 minutes watching to a recording, so I speeded it a little up and deleted some unwanted scenes in the video below.

Ok. At this point you will see the setup page of your just installed virtual MTR. You can accept the terms and configure your account, but probably you will end up at this screen:

So, we are going to exit this setup, by going back to the **EULA screen** and press **Exit**. At the user screen, login as the local administrator. Default password for the local administrator is “sfb”. Please change this!

Browse to the following folder:

C:\Users\Skype\AppData\Local\Packages\Microsoft.SkypeRoomSystem\_8wekyb3d8bbwe\LocalState

and create a new *SkypeSettings.xml*-file with the following settings:

<SkypeSettings>

<UserAccount>

<SkypeSignInAddress>test@test.com</SkypeSignInAddress>

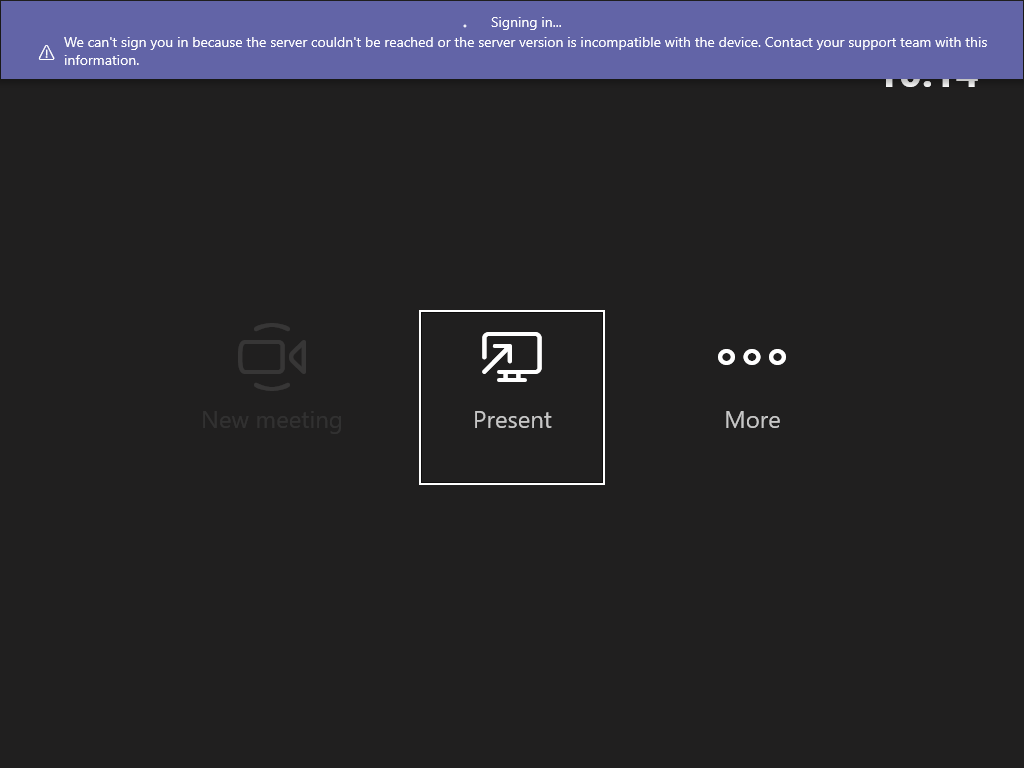
<ExchangeAddress>test@test.com</ExchangeAddress>

<Password>test</Password>

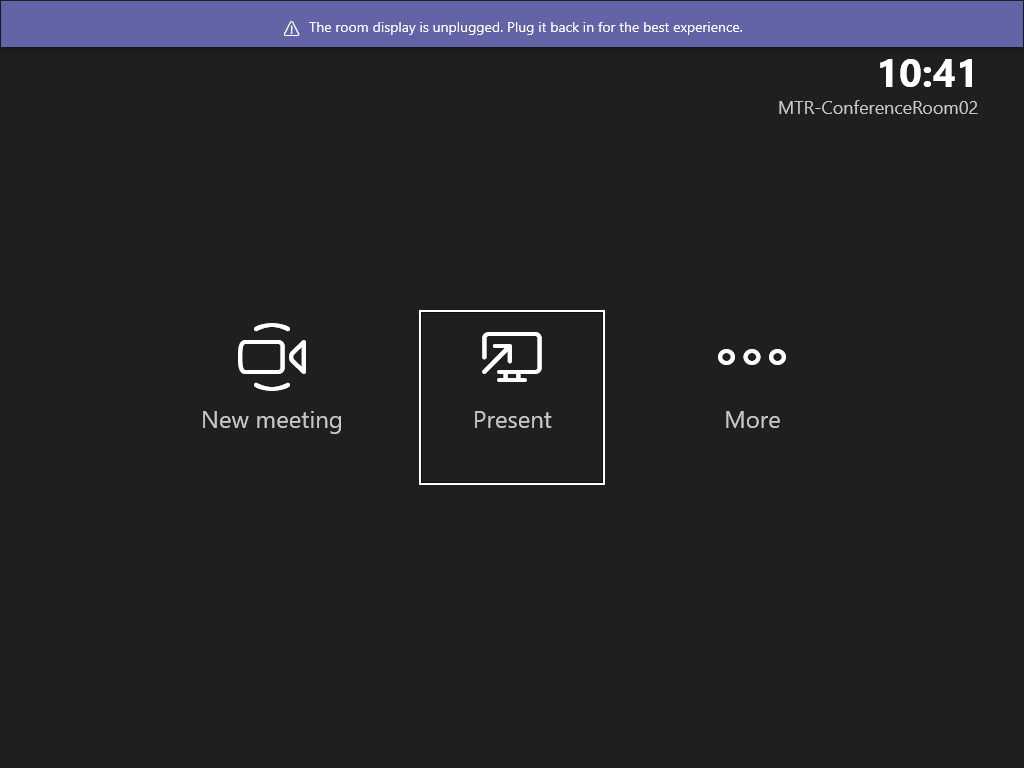
</UserAccount>

</SkypeSettings>

Copy

All set? Reboot the MTR and hopefully the MTR will pick up the XML file and configure the fictive account. You will see that the account cannot sign in. That is correct.

Click on the three dots (**More**), click **Settings**, fill in the local administrator password. Go to **Account** and fill in your corresponding account settings which you created in part 1 and click **Save & exit**.

As an end result, you will see the below screen. The account is succesfully signed-in, but the MTR want's to have a external display for the best experience. Ignore that. 😀  


That is, it for now. Next part will be to connect the MTR with Endpoint Manager so we can manage this device.

**Resources**  
• <https://docs.microsoft.com/en-us/microsoftteams/rooms/rooms-lifecycle-support>  
• <https://go.microsoft.com/fwlink/?linkid=851168>  
• <https://docs.microsoft.com/en-us/microsoftteams/rooms/xml-config-file>

**Gettings started with Microsoft Teams Rooms (Part 3)**

by [Jeroen Burgerhout](https://burgerhout.org/author/jeroen/) 5 months ago  4 MIN READ

Welcome back for part 3 of this series on how to get started with Teams Rooms.

In part 1, we have configured the MTR-account in Office365, assigned a Meeting Room license, created a resource mailbox with some optional parameters for automatic reply and so on, enabled Skype for Business (as a room) and set the password no not expire.

In part 2, we have installed an MTR in Hyper-V and installed the MTR software on the device. We also configured the MTR with the account settings from part 1.

This series is divided in the following posts:  
Part 1: [Creating the Teams Room account and arrange the license](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-1/)  
Part 2: [Installing Microsoft Teams Room in Hyper-V](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-2/)  
Part 3: [Connect and manage it with Microsoft Endpoint Manager](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-3)  
Part 4: [Customize the Microsoft Teams Room](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-4/)  
Part 5: ?

You can find the files, scripts and other stuff that I have used in this series on my GitHub page at <https://github.com/BurgerhoutJ/scripts/tree/main/microsoft-teams-rooms>.

**Part 3: Connect and manage it with Microsoft Endpoint Manager**

In this part, I assume that you have followed part 1 and part 2, we have a working MTR available. You can check the compliance of the device with Microsoft Endpoint Manager by pushing a compliance policy. You can also push other configurations, but the MTR handles it’s own updates.

**What do we need?**

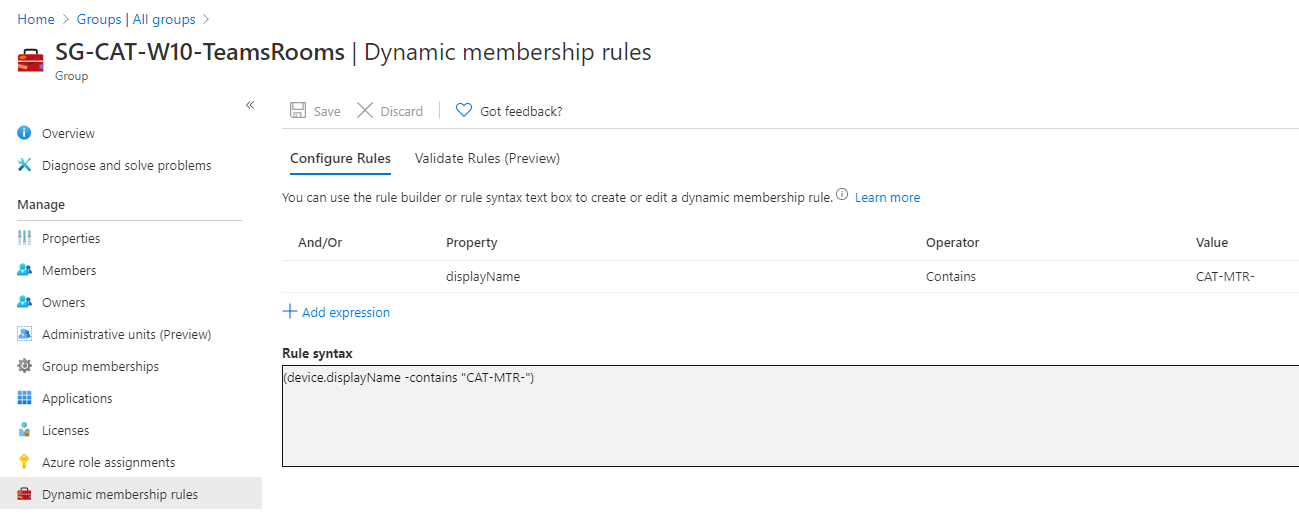
You must have the following:

* A working Microsoft Teams Room
* A Microsoft Endpoint Manager / Azure AD tenant
* Internet (of course)
* An Azure AD Device Group
* Compliance policy

**Azure AD Device Group**

First, we are going to create a device group in Azure AD to populate all the MTR’s into one group.

To do this, go to [https://endpoint.microsoft.com](https://endpoint.microsoft.com/) and login and proceed with the following steps:

1. **Go** to **Groups**
2. **Click** on **New group**
3. Give the group a name, in my case the group is called ‘SG-CAT-W10-TeamsRooms’
4. Set the Membership type to **Dynamic Device**
5. **Click** on **Add dynamic query**
6. And fill in the Rule syntax field the following syntax: *(device.displayName -contains "CAT-MTR-")*. Change the *CAT-MTR* to your device naming template.
7. At the end, the group should like this: 

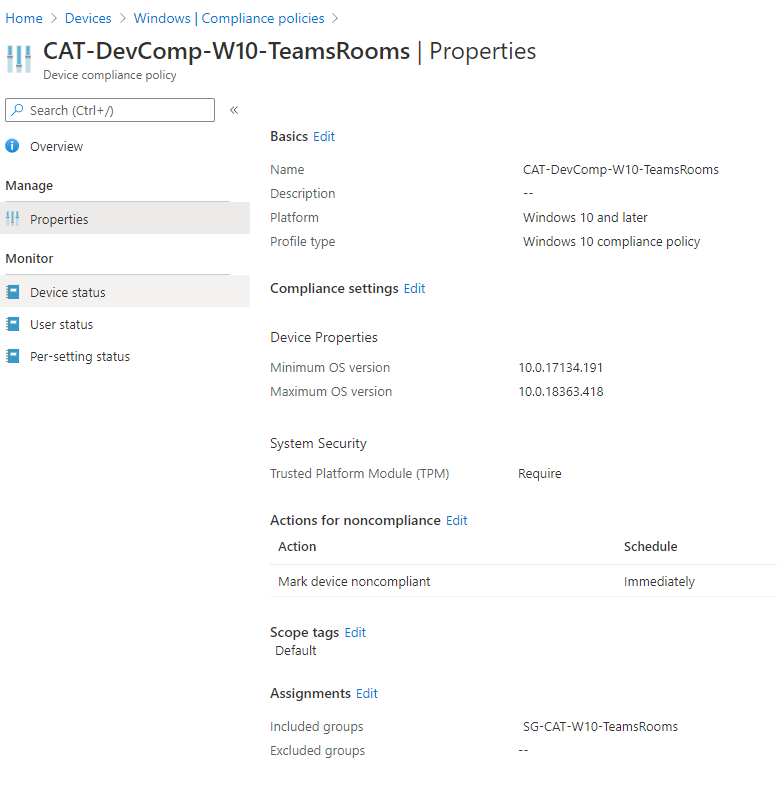
That is, it for the Azure AD group. Next, we go to create a Compliance policy.

**Compliance policy**

To get an insight if your active Microsoft Teams Rooms are still compliant, you can create a Compliance policy. With this Compliance policy, you can set for example, the minimum and the maximum OS version, if the MTR requires a TPM and so on.

To create a Compliance policy, go to **Devices** -> **Windows** -> **Compliance policies**.

1. **Click** on **Create Policy**
2. Select **Windows 10 and later** as Platform and click on **Create**
3. Give the policy a name and click on **Next**
4. Under the **Device Properties** node, type *10.0.17134.191* as minimum OS and *10.0.18363.418* as maximum OS.
5. Under **System Security**, require a Trusted Platform Module (TPM)
6. For now, click **Next**
7. Click **Next** again
8. Assign this policy to the Azure AD group, which you created earlier, and click **Next**
9. And at the **Review + create** tab, review everything and click on **Create**.



Now we have created a Compliance policy and assigned it to the earlier created Azure AD group. Next up, the Device Enrollment Manager.

We are switching to the MTR and we are going to connect the MTR to your AzureAD tenant, and we can manage this through the Endpoint Manager portal.

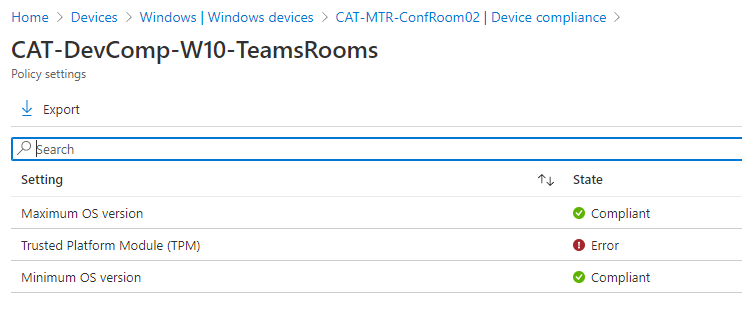
On your Microsoft Teams Room, exit the Microsoft Teams Room environment and login with the local Administrator account.

1. Open **Settings**
2. **Click** on **Accounts**
3. **Click** on **Enroll only in device management**
4. Type the email address of your resource account (MTR account)
5. Provide the necessary password
6. **Click** on **Got it**

**TIP: Check after a while if the folder *C:\Program Files (x86)\Microsoft Intune Management Extension*. If it is not there (yet), you can manually install the Intune Agent by downloading it from**[**https://burgerhou.tj/IntuneWindowsAgent**](https://burgerhou.tj/IntuneWindowsAgent)**.**

The MTR is now connected with your Microsoft Endpoint Manager / Azure AD tenant.

After a few minutes, you will see in the MEM portal, that your device is compliant or not. In my case not, I see, because I forgot to enable TPM on the VM.



And with this, we are at the end of this part. In the next part, we will look on how we can customize the MTR for example a custom theme, so see you next time!

**Resources**

* <https://techcommunity.microsoft.com/t5/intune-customer-success/managing-teams-meeting-rooms-with-intune/ba-p/1069230>
* <https://docs.microsoft.com/intune/enrollment/device-enrollment-manager-enroll>

**Gettings started with Microsoft Teams Rooms (Part 4)**

by [Jeroen Burgerhout](https://burgerhout.org/author/jeroen/) 4 months ago  5 MIN READ

It is time for part 4 of this series about Microsoft Teams Rooms.

In part 1, we have configured the MTR-account in Office365, assigned a Meeting Room license, created a resource mailbox with some optional parameters for automatic reply and so on, enabled Skype for Business (as a room) and set the password no not expire.

In part 2, we have installed an MTR in Hyper-V and installed the MTR software on the device. We also configured the MTR with the account settings from part 1.

In part 3, we connected the MTR with Intune to check some compliancy and to manage your Teams Room.

This series is divided in the following posts:  
Part 1: [Creating the Teams Room account and arrange the license](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-1/)  
Part 2: [Installing Microsoft Teams Room in Hyper-V](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-2/)  
Part 3: [Connect and manage it with Microsoft Endpoint Manager](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-3)  
Part 4: [Customize the Microsoft Teams Room](https://burgerhout.org/gettings-started-with-microsoft-teams-rooms-part-4/)  
Part 5: ?

You can find the files, scripts and other stuff that I have used in this series on my GitHub page at <https://github.com/BurgerhoutJ/scripts/tree/main/microsoft-teams-rooms>.

**Part 4: Customize the Microsoft Teams Room**

In this part of the series, we are going to customize your MTR with for example a custom theme that suits your organization. I will show you also, how you can manage some basic settings with the help of the Teams Admin Center.

**What do we need?**

You must have the following:

* A custom background
* Access to Endpoint Manager and Teams Admin Center

**Create an XML config file**

To customize your MTR, you need to create an XML file, called SkypeSettings.xml. We did this already in part 2. With this XML file, you can customize many things, like default external devices to use, like microphones and cameras. You can configure to send logs to an email address, and much more settings. In the Resources section, you will find the link to all the settings that you can configure.

For this blog post, we are going to concentrate to the following XML configuration:

<SkypeSettings>

<Theming>

<ThemeName>Custom</ThemeName>

<CustomThemeImageUrl>CustomBackgroundTheme.png</CustomThemeImageUrl>

<CustomThemeColor>

<RedComponent>249</RedComponent>

<GreenComponent>148</GreenComponent>

<BlueComponent>59</BlueComponent>

</CustomThemeColor>

</Theming>

</SkypeSettings>

Copy

**Custom Theme**

Before you begin with your custom theme, please note of the following prerequisites:

* Image must be exactly 3840x1080 pixels
* Must be a jpg, jpeg, png or bmp file format
* A graphic designer can use the provided [Photoshop template](https://docs.microsoft.com/en-us/microsoftteams/downloads/ThemingTemplateMicrosoftTeamsRooms_v2.1.psd)

The image can be used for dual screen setup. So, keep in mind that you can divide your image in a left and right section. I created one big image, and this is for testing only. Maybe I am going to setup a NUC with dual screen for a result.



Nice feature that I found on the internet, is from [@patrichard](https://twitter.com/patrichard/). He created a PowerShell script that change your background image every day. Link to his blogpost is available in the Resources section.

**XML file configuration**

Now we have a custom image, and we need to store it on a central space, so your MTR can access it. I will put it on my website. Let us look at the XML configuration file.

<SkypeSettings>

<Theming>

<ThemeName>Custom</ThemeName>

<CustomThemeImageUrl>CustomBackgroundTheme.png</CustomThemeImageUrl>

<CustomThemeColor>

<RedComponent>249</RedComponent>

<GreenComponent>148</GreenComponent>

<BlueComponent>59</BlueComponent>

</CustomThemeColor>

</Theming>

</SkypeSettings>

Copy

Save the above XML as SkypeSettings.xml and store this file on the same location as your image.

**Creating the PowerShell script**

Right now, we have a custom image, an XML file with the settings. But how can we configure the MTR so that he is going to use your custom theme.

To do this, we need a PowerShell script and Microsoft Endpoint Manager.

Thanks to Thomas Eklund, we can use his PowerShell script and do some modifications to it.

$WebClient = New-Object System.Net.WebClient

$WebClient.DownloadFile("<https://yourdomain.com/CustomBackgroundTheme.png>","C:\Users\Skype\AppData\Local\Packages\Microsoft.SkypeRoomSystem\_8wekyb3d8bbwe\LocalState\CustomBackgroundTheme.png")

$WebClient = New-Object System.Net.WebClient

$WebClient.DownloadFile("<https://yourdomain.com/SkypeSettings.xml>","C:\Users\Skype\AppData\Local\Packages\Microsoft.SkypeRoomSystem\_8wekyb3d8bbwe\LocalState\SkypeSettings.xml")

if (-not (Test-Path -LiteralPath 'c:\temp')) {

New-Item -Path 'c:\temp' -ItemType Directory

}

else {

"Directory already existed"

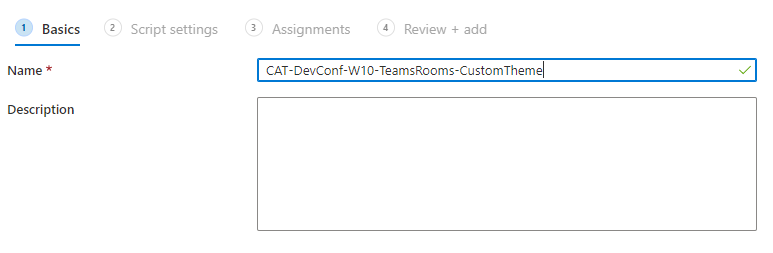
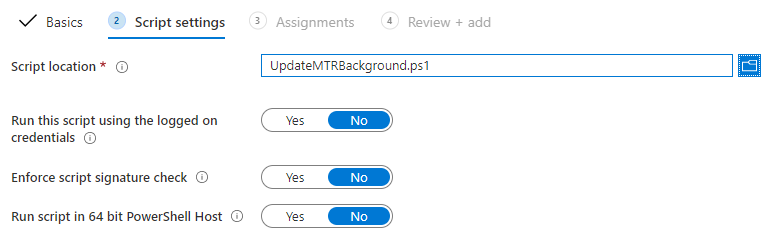
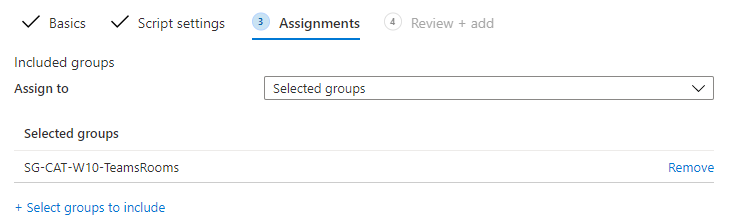
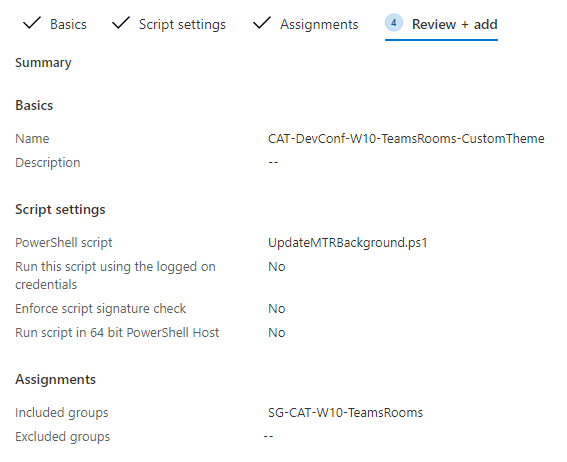
}

Copy

The above PowerShell script will download the image and the xml file and place it in the correct folder. Save the PowerShell script.

**Microsoft Endpoint Manager**

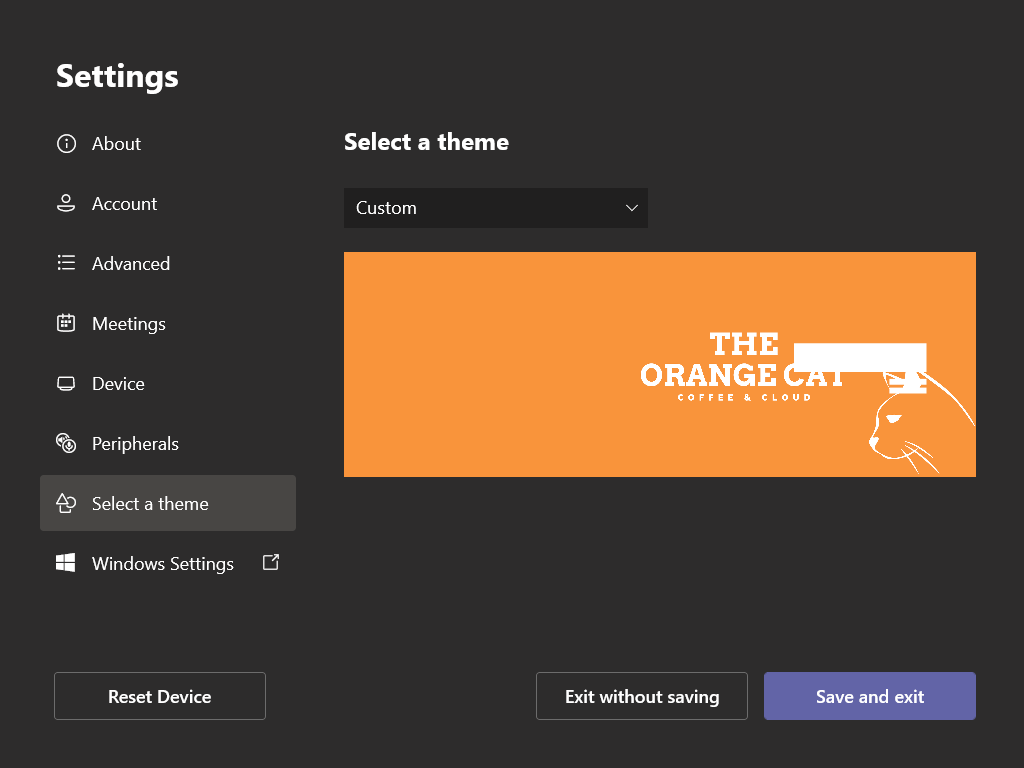
Open your MEM portal, logon and go to **Devices** -> **Windows** -> **PowerShell scripts**.

1. Click on **+** **Add**
2. Give it a name and click **Next**
3. Browse to your PowerShell script and click **Next**
4. Assing the script to your MTR AzureAD group and click **Next**
5. Review your settings and click on **Add**

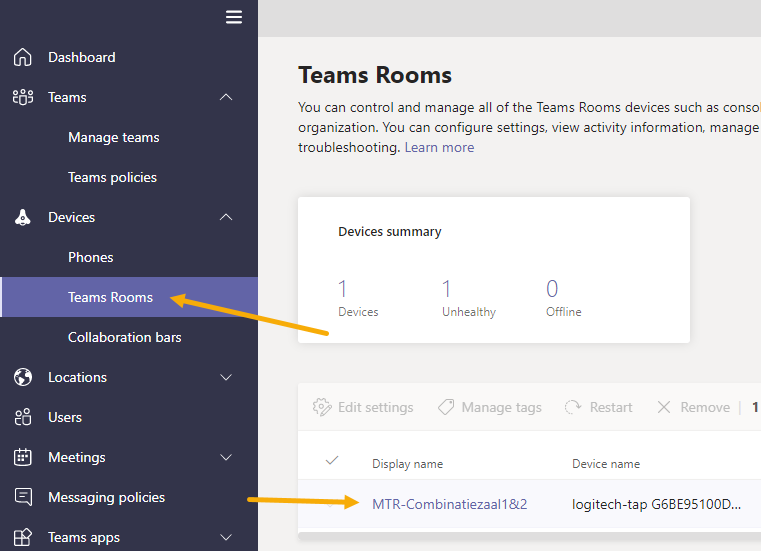
And now we must wait until the MTR is contacting Intune and execute the PowerShell script.

**….2 hours later….**

My MTR has received the PowerShell script and the new custom-made theme, as shown in the screenshot below.



Of course, you can configure many things with the SkypeSettings.xml file, but nowadays you can manage your MTR with the Teams Admin Center. In the following screenshot, you see a Teams Room device in the TAC.



Hope you liked this part of my series. If you have some ideas for part 5, then let me know.

For now, have a good day.

**Resources**

* <https://docs.microsoft.com/en-us/microsoftteams/rooms/xml-config-file#create-an-xml-configuration-file>
* <https://docs.microsoft.com/en-us/microsoftteams/rooms/xml-config-file#custom-theme-images>
* <https://www.ucunleashed.com/4323>
* <https://www.linkedin.com/pulse/modern-deployment-teams-meeting-room-settings-thomas-eklund/>
* <https://docs.microsoft.com/en-US/microsoftteams/rooms/rooms-manage>