Meeting4Display

Microsoft Graph

Workspace management solution

CONFIGURATION GUIDE

V3.2

https://www.telelogos.com/en/solutions/meeting4display-workspace-management-solution/

1. Introduction

The Meeting4Display solution uses Microsoft Graph APIs.

2. Office 365 mailbox configuration

The mailbox is configured via PowerShell using the **"Exchange Online PowerShell for MFA"** command line interface.

You must install the tool and execute the commands via PowerShell using your Office 365 administrator account.

Use MS-Edge to download this tool.

This tool is available from the Exchange admin center https://outlook.office365.com/ecp/. Under the Hybrid menu, select "Configure" in the Exchange Online PowerShell section to download and install the module that supports multi-factor authentication. The commands will then be executed from this module. The module can be installed on any workstation that has access to the Office 365 server.

We recommend creating distribution list groups to manage configuration of the resources used in the Meeting4Display application.

"Rooms" resources can be added to a "Rooms" distribution group and "Desks" resources can be added to an "Desks" distribution group.

Note: Items noted in red italics in the PowerShell commands to be executed are examples only and should be replaced with your own values.
All these commands are available on the Microsoft website.

Connect to Exchange Online PowerShell using multi-factor authentication:

Login (with an Administrator account)
Connect-EXOPSSession -UserPrincipalName *admin@mydomain* ➔ *Login with an Administrator account.*

Creating a room mailbox:

Create
New-Mailbox –Name "*Room_name*" –PrimarySmtpAddress *room_name@mydomain* –Room –EnableRoomMailboxAccount $true –RoomMailboxPassword (ConvertTo-SecureString -String *Password* -AsPlainText -Force)

Verify
Get-Mailbox -Identity "*Room_name*" | Format-List Name,DisplayName,Alias,PrimarySmtpAddress,Database

Creating a resources list:

Create
New-DistributionGroup -Name "*Room_list_name*" -RoomList

Verify
Get-DistributionGroup -Identity "*Room_list_name*" | Format-List

Adding a resource to the resources list:

Add
Add-DistributionGroupMember -Identity "*Room_list_name*" -Member "*room_name@mydomain*"

Verify
Get-DistributionGroupMember -Identity "*Room_list_name*"

Configuration of resource options (rooms/desks):

Common options:

Resource option settings can be assigned individually to each resource or all the resources linked to a resource list (*Distribution Group*).

The options required for Meeting4Display to function properly are as follows:

- **DeleteComments** allows you to indicate that the body text of incoming booking request messages should be saved.
- **RemovePrivateProperty** specifies that you should not clear the private flag for incoming bookings sent by the host in the original requests.
- **DeleteSubject** indicates that the subject of incoming booking requests should be saved.
- **AddOrganizerToSubject** specifies that the name of the booking organizer is not used as the subject of the booking request.
- **AutomateProcessing** enables the processing of calendar items in the mailbox. This means that the Calendar Wizard updates the calendar and the Resource Reservation Wizard accepts the meeting according to the policies.

Add room

```
Set-CalendarProcessing
-Identity "Room_name"
-DeleteComments $false
-RemovePrivateProperty $false
-DeleteSubject $false
-AddOrganizerToSubject $false
-AutomateProcessing AutoAccept
```

Verify

```
Get-CalendarProcessing
-Identity "Room_name" | Format-List
```

Add list

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
  Set-CalendarProcessing
  -Identity $_.Identity
  -DeleteComments $false
  -RemovePrivateProperty $false
  -DeleteSubject $false
  -AddOrganizerToSubject $false
  -AutomateProcessing AutoAccept
}
```

Verify

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
  Get-CalendarProcessing
  $_.Identity | Format-List
}
```

User-dependent options (to be configured for Meeting4Mobile):

Meeting4Mobile requires additional room rights to be configured on resources for users so that they can access information about the booking of the resource (e.g. *Display room status in "My bookings" and manage desk confirmation*).

Note

If the Calendar directory does not exist for the resource, it may exist under the calendar name.

Reviewer permission must be applied at least to resource calendars. If noneditingauthor permission is already set, it can be kept.

Add room

```
Add-MailboxFolderPermission
-Identity Room_address:\calendar
-User "By default"
-AccessRights Reviewer
```

Verify

```
Get-MailboxFolderPermission
-Identity Room_address:\calendar
```

Add list

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
  Add-MailboxFolderPermission
  -Identity "$($_.Identity):\calendar"
  -User "By default"
  -AccessRights Reviewer
}
```

Note

If the Add-MailboxFolderPermission command does not work because permissions are already configured, use the Set-MailboxFolderPermission command to manage the permissions:

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
    Set-MailboxFolderPermission
    -Identity "$($_.Identity):\calendar"
    -User "By default"
    -AccessRights Reviewer
}
```

Verify

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
  Get-MailboxFolderPermission
  "$($_.Identity):\calendar" | Format-List
}
```

3. Azure Portal configuration

3.1 Configuration of the application

Meeting4Display uses OpenID authentication via Microsoft Authentication Library (MSAL).

This authentication system requires configuration on the Microsoft Cloud Services management website.

Account to use	Website
Administrator account	https://portal.azure.com/

Once logged in:
- Find and open the "App Registrations" portal.
- Click on "New registration".
- Enter the application name (e.g. *Meeting4Display*):
    - Select the option "*Accounts in this organizational directory only (MyCompany only – Single-tenant)*"
    - Define the redirect URL as Web.
      The redirect URL must have the following format:
      Https://{nomhote}/Meeting4DisplayMobile/
    - When finished, select "Register".

Once the application is registered, **copy** and **keep** the application identifiers (application (client) ID) and the directory (directory (tenant) ID).

Then go to the "Authentication" menu of the application:
- Add the following default options:
    - Access tokens;
    - ID tokens.
- Then save.

Go to the "API permissions" menu:

- Click on "Add a permission"

- Search for and select the Microsoft Graph API:
    - Select "**Application permissions**":
        - Find and click on "Place"
            - Select Place.Read.All
        - Find and click on "Calendars"
            - Select Calendars.ReadWrite
              It is possible to select Calendars.Read but some features listed in paragraph 2 1.1 will not be possible.
        - Find and click on "Mail"
            - Select Mail.Send
    - Click on "Add permissions";
    - Click on "Grant admin consent for…".
- Click on "Add a permission"

- Search for and select the Microsoft Graph API:
    - Select "**Delegated permissions**":
        - Find and click on "Calendars"
            - Select Calendars.ReadWrite
            - Select Calendars.Read.Shared
        - Click on "Add permissions"

Finally go to the "Certificates & Secrets" menu
- Click on "New customer secret"
- Add a description and an expiration date, then click on "Add".
    - **Copy** the secret client value and **store** it with the previously obtained identifiers.

3.2 Permissions used by Meeting4Display

"Application" permissions

- Place.Read.All
    - [BackOffice] API connection test (resource retrieval)
    - [BackOffice] Retrieve the list of resources (room/desk) in the console for adding
    - [Widget] Retrieve the list of resources for selection

- Calendars.Read
    - [Meeting4Room] Retrieve meetings
    - [Meeting4Room] Search for a free room
    - [Meeting4Outlook] Search for a room/a desk
    - [Widget] Retrieve information to display widgets
    - [BackOffice] Retrieve information for statistics and reports

- Calendars.ReadWrite
    - [Meeting4Room] Action on meetings (Create/Extend/End/Cancel)
    - [Meeting4Mobile] Manage automatic confirmation for FlexOffice

- Mail.Send
    - [Meeting4Room] Send the email for the concierge
    - [BackOffice] Send reports

"Delegated" permissions

- Calendars.ReadWrite
  - [Meeting4Mobile] Search for and book a room/a desk
  - [Meeting4Mobile] Scan and book a desk
  - [Meeting4Mobile] Display bookings
  - [Meeting4Mobile] Action on bookings (Extend/End/Cancel)

- Calendars.Read.Shared
  - [Meeting4Mobile] Display room status of the user's bookings
  - [Meeting4Mobile] Manage desk confirmation

Restriction of app permissions

It is possible to restrict application access to specific mailboxes by creating an application access policy:

https://docs.microsoft.com/en-us/graph/auth-limit-mailbox-access

4. Meeting4Display Back Office

The application can be accessed here:
http(s)://{hostnameIPaddress}/Meeting4Display

The settings to use the application, created in the Azure portal, for the Meeting4Display suite are configured from the "Calendar Configuration" tile under the "Settings" menu or the "Settings" tile on the "Home" page.

- The elements to be defined are as follows:

Calendar system	Office 365 (Microsoft Graph)
Client ID	Application (client) ID obtained when configuring the application on the Azure portal
Tenant ID	Directory (tenant) ID obtained when configuring the application on the Azure portal
Client Secret	Client secret obtained when configuring the application on the Azure portal

- The "Test" button is used to check that the Meeting4Display application communicates correctly with the Graph APIs.

5. Meeting4Mobile

The application can be accessed here:
http(s):// {hostnameorIPaddress}/Meeting4DisplayMobile

The Meeting4Mobile application uses basic authentication. It allows you to connect with a user account (login/password) defined in Office 365.
The Meeting4Mobile application uses "delegated" permissions for Microsoft Graph APIs.

6. Outlook Add-in

The application can be accessed here:
http(s)://{{{hostnameorIPaddress}/Meeting4DisplayOutlook/

The Outlook Add-in application uses permissions "applications" for Microsoft Graph APIs.

To be able to use it, you have to enter the URL and password needed to login to the company, then download the XML configuration file and add it to the Outlook Add-ins.