Meeting4Display

Office 365

Workspace management solution

CONFIGURATION GUIDE

V3.2

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

1. Introduction

The Meeting4Display solution uses Office 365 Exchange Web Services (EWS).

This service must be enabled and accessible.

The Client Access Server (CAS) role must be enabled to allow a third-party application to connect to it.

2. Office 365 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 Internet Explorer or Edge to download this tool.

The tool can be downloaded 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.

The Meeting4Display suite uses a service account to perform actions on Exchange.

Available rooms are managed via the room list functionality in the application (Exchange RoomLists).

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.

Obtaining the type of license to assign to the service account:

Obtaining available licenses (using an administrator account)
Connect-MsolService ➔ *Login with an Administrator account.* Get-MsolAccountSku ➔ *Copy the AccountSkuId of the license type you wish to assign to your service account (e.g. mydomain:O365_BUSINESS_ESSENTIALS)*

Creating a service account:

Create
New-MsolUser -DisplayName *"Meeting4Display Service Account"* -FirstName *"Account Service"* -LastName *"Meeting4Display"* -UserPrincipalName *serviceaccount@mydomain* -LicenseAssignment *mydomain:O365_BUSINESS_ESSENTIALS* • *Paste the AccountSkuId you copied in the previous step.*

Verify
Get-MsolUser -UserPrincipalName *serviceaccount@mydomain*

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 room list:

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

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

Adding a room to the room list:

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

Verify
Get-DistributionGroupMember -Identity *"Room_list_name"*

Assigning delegation permissions to the service account:

Room delegation permissions can be assigned to the service account for a single room or all the rooms in a room list (RoomList).

The "full access" permission setting is assigned to the service account.

Add room

```
Add-MailboxPermission
-Identity "Room_name"
-User "Account_Name"
-AccessRights FullAccess
```

Verify

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

Add list

```
Get-DistributionGroupMember
-Identity "Room_list_name" |
ForEach-Object
{
  Add-MailboxPermission
  $_.Identity
  -User "Account_Name"
  -AccessRights FullAccess
}
```

Verify

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

Configuring the service account for impersonation:
Impersonation can be performed:

- By using a role group which the service account is a member of
- By associating the role directly to the service account

Using a role group

By creating a role group allowing impersonation and containing the service account as a member, impersonation can be configured for the service account in a single command (New-RoleGroup).

Note: To add impersonation permissions to an existing group, use the command Set-RoleGroup. Then, to add the service account to the group, use the command Update-RoleGroupMember.

Add
New-RoleGroup -Name *"Group_Name"* -Roles ApplicationImpersonation -Members *"Account_Address"*

Verify
Get-RoleGroup -Identity *"Group_name"* Get-RoleGroupMember -Identity *"Group_name"*

Adding the role directly to the service account (without adding a role group)

Add
New-ManagementRoleAssignment -Name *"Role_Name"* -Role ApplicationImpersonation -User *"Account_Address"*

Verify
Get-ManagementRoleAssignment *"Role_Name"*

Configuring the room options:

Common options:

Room option settings can be assigned to the service account for a single room or all the rooms in a room list (RoomList).

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

- **DeleteComments** allows you to indicate that the body text of incoming meeting request messages should be saved.
- **RemovePrivateProperty** specifies that you should not clear the private flag for incoming meetings sent by the host in the original requests.
- **DeleteSubject** indicates that the subject of incoming meeting requests should be saved.
- **AddOrganizerToSubject** specifies that the name of the meeting organizer is not used as the subject of the meeting 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 for users.

Add room

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

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 noneditingauthor
}
```

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 noneditingauthor
}
```

Verify

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

3. Meeting4Display Back Office

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

The back office 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) and define the redirect url as Web.
The redirect URL must have the following format:
Https://{hostname}/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 and add the following default options:

- Access tokens;
- ID tokens.

Then save.

Go to the "API permissions" menu and click on "Add a permission".
Search for the Exchange API:

- Select « APIs my organization uses » and search « Office 365 Exchange Online »
- Select "Delegated permissions" and then select "EWS.AccessAsUser.All" under EWS. Click on "Add permissions".
- Then select "Application permissions" and under "Permissions" check "full_access_as_app", then click on "Add permissions".

Click on "Grant admin consent for…".

Finally, go to the "Certificates & Secrets" menu and 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.

The settings to use the service account 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 (EWS)
Server address	https://outlook.office365.com/ews/exchange.asmx
Account name	Service account name: Example: *Meeting4Display@domain.onmicrosoft.com*
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 Exchange Web Services (EWS).

4. 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.

5. Outlook Add-in

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

The Outlook Add-in application uses authentication via MSAL with "impersonation". It allows you to log on with the service account properties configured in the back office.

In order to use it, you must enter the URL and password required to log on to the company.