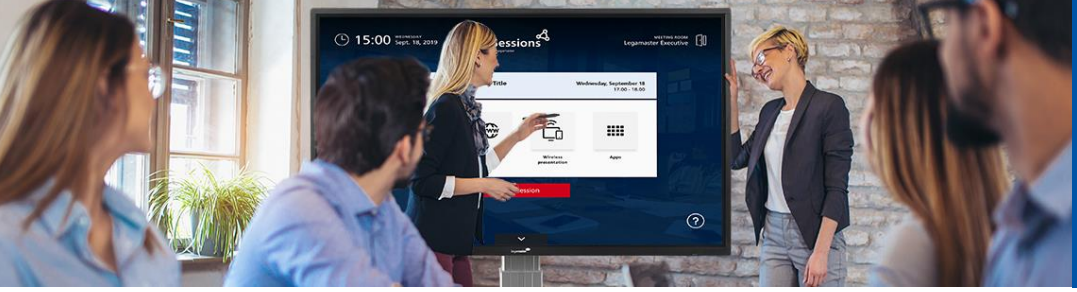


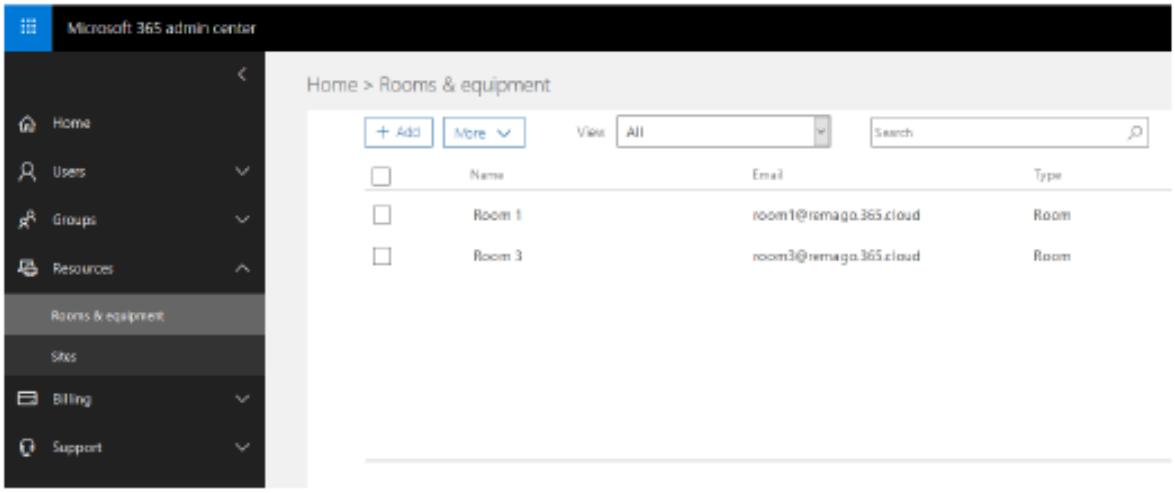
Sessions
By Legamaster

SESSIONS SOFTWARE

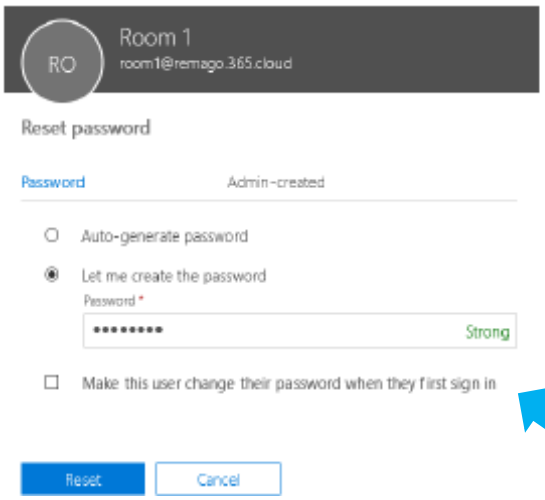
EXCHANGE SERVER CONFIGURATION IN OFFICE 365

Create a Meeting Room resource account

1. Access Microsoft 365 admin center.
2. Navigate to Resources > Rooms & equipment.
3. Click [+Add] button and follow the on-screen instructions to create a meeting room resource account.



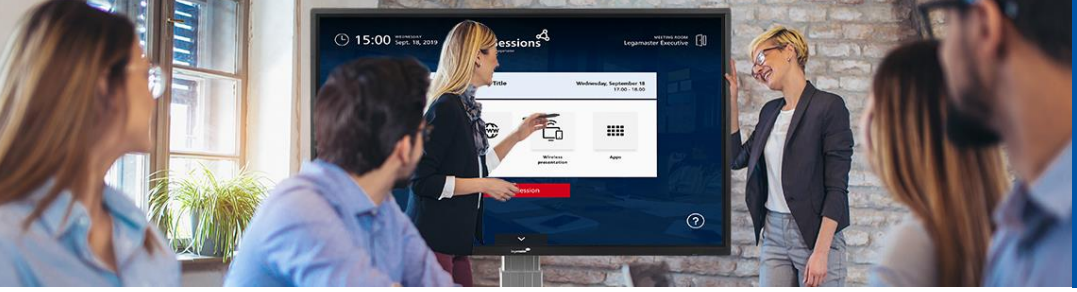
4. Navigate to User > Active Users and the room you just created.
5. Click the [Reset password] button and create a strong password for the account.



Clear the **Make this user change their password when they first sign in** check box.

6. If you need to log in from the meeting room account to **Skype for Business**, activate the related license for the resource through the **Product licenses** option.

The room resource account should have the default lowest access rights level.
The account does not need to be added to special groups or be assigned a specific role.



Now you must open a **PowerShell** console and start a **Microsoft Online Service** session to execute the commands listed below. For instructions on how to configure a client to administrate Office 365, see Appendix A.

Show the meeting subject in the Launcher's calendar

By default, the meeting title (Subject) contains the name of the organizer.
You need to get the real meeting subject.

```
Set-CalendarProcessing -Identity "Room ID" -AddOrganizerToSubject 0

Set-CalendarProcessing -Identity "Room ID" -DeleteSubject 0
```

Create a meeting room resource list

```
New-DistributionGroup -Name "Room List ID"
-DisplayName "Room List Friendly Name"
-PrimarySmtpAddress roomlist1@domain.com -RoomList
```

Add the new meeting room resource account to the meeting room resource list

```
Add-DistributionGroupMember -Identity "Room List ID"
-Member "Room ID"
```

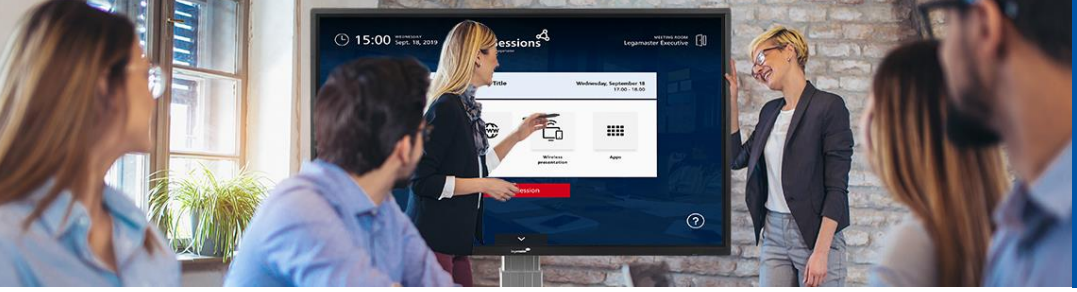
To execute the above commands, you may need to execute the following commands.

Get a list of all meeting room resources

```
Get-Mailbox | Where-Object {$ .RecipientTypeDetails
-eq "RoomMailbox"} | Format-Table
DisplayName,Identity,PrimarySmtpAddress
```

Get a list of all meeting room resources lists

```
Get-DistributionGroup | Where {$_.RecipientTypeDetails
-eq "RoomList"} | Format-Table
DisplayName,Identity,PrimarySmtpAddress
```



Sessions
By Legamaster

APPENDIX A

Administering Office 365 Exchange through PowerShell

Office 365 PowerShell lets you to manage your Office 365 settings from the command line. Connecting to Office 365 PowerShell is a simple process where you install the required software and then connect to your Office 365 organization. Use one of the following 64-bit versions of Windows:

- Windows 10+, Windows 8.1, Windows 8 o, Windows 7 Service Pack 1 (SP1) or
- Windows Server 2019+, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 SP1.

You need to install on your local computer the **Microsoft Online Services Sign-in Assistant**. To do so, open an elevated Windows PowerShell command prompt (run Windows PowerShell as an administrator) and run the following command.

```
Install-Module MSOnline
```

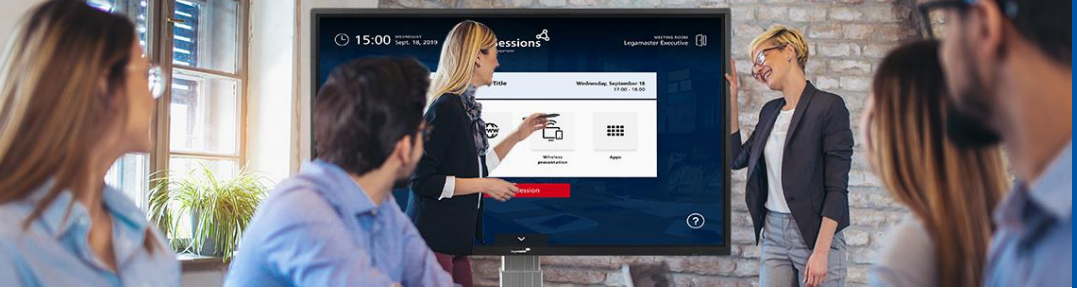
If prompted to install the **NuGet** provider, type Y and press ENTER.
If prompted to install the module from **PSGallery**, type Y and press ENTER.
If your PowerShell is not enabled to execute scripts that are not digitally signed, run the following command:

```
Set-ExecutionPolicy RemoteSigned
```

Open an admin session

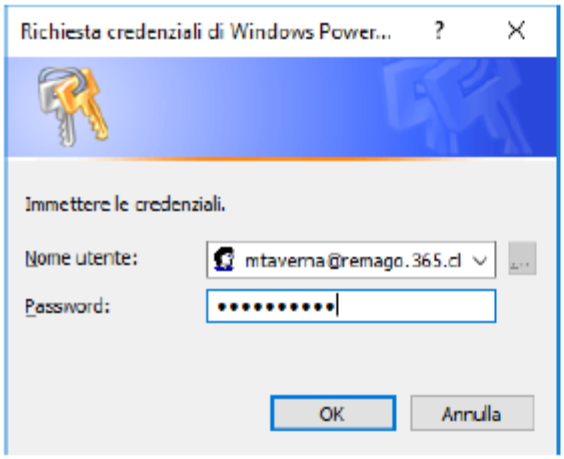
Run the following command:

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



Sessions
By Legamaster

Insert the credentials of a user who is member of an Office 365 admin role:

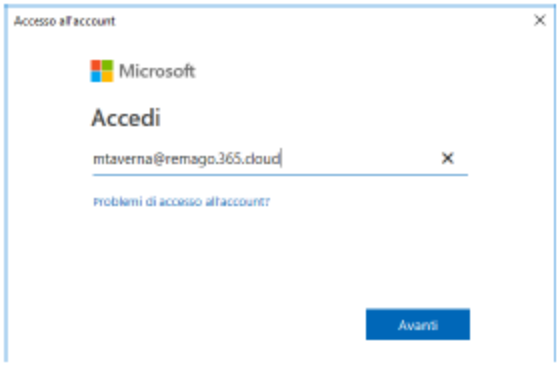


Run the following commands:

```
Import-PSSession $Session

Connect-MsolService
```

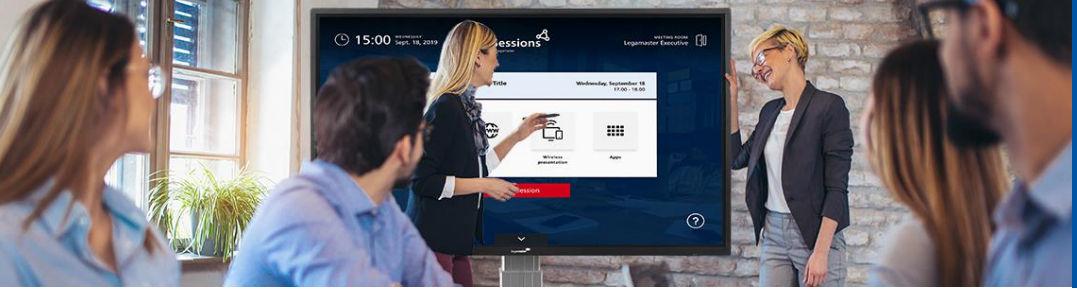
Insert the credentials of the Office 365 admin user again:



To verify if the session is active and running, you can run these commands:

```
Get-MsolUser

Get-Mailbox
```



Close an admin session

When finished, remember to close the remote PowerShell session:

```
Remove-PSSession $Session
```

By closing the PowerShell window without disconnecting the session, you risk running out of available PowerShell remote sessions and having to wait for sessions to expire.

Useful commands

To get the complete history of commands you executed:

```
Get-History
```