

## Office 365 PowerShell Workshop – Hands on labs

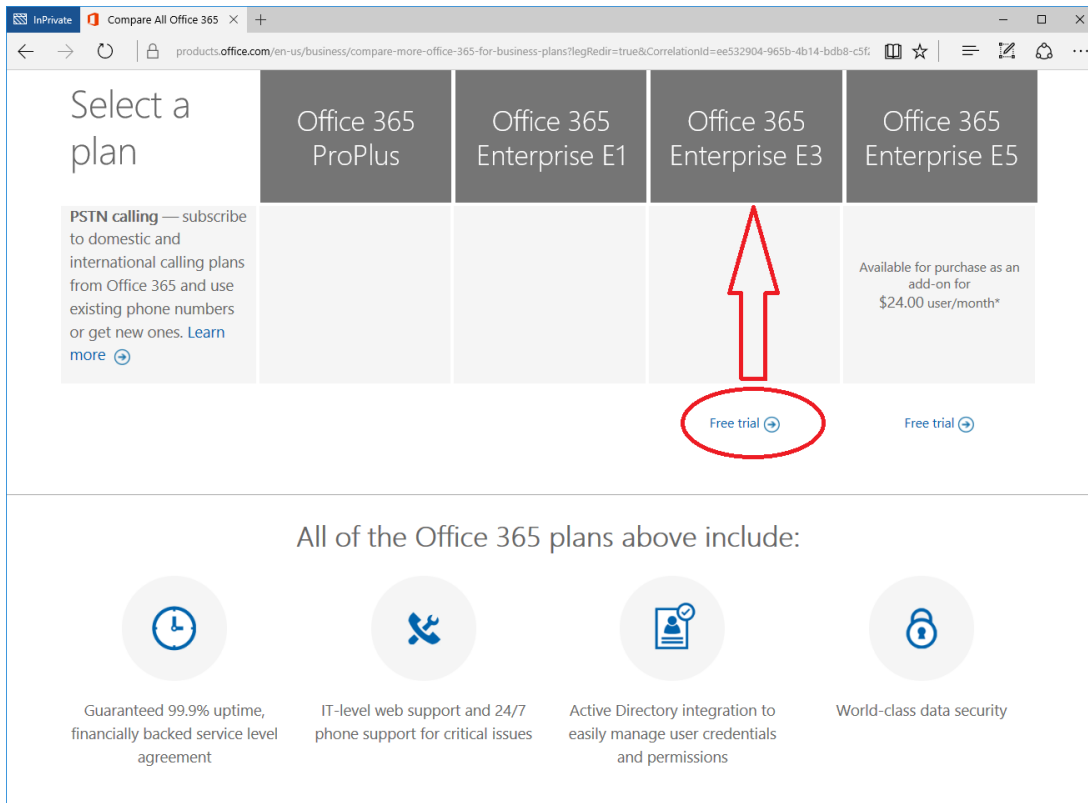
We are looking forward to having you in our Office 365 PowerShell workshop on Monday October 10<sup>th</sup> at the IT/Dev Connections 2016 preconference workshops.

To prepare yourselves for the hands-on part of the workshop we would like to ask you to register for a test tenant in Office 365 so that you don't have to play around in your production environment 😊

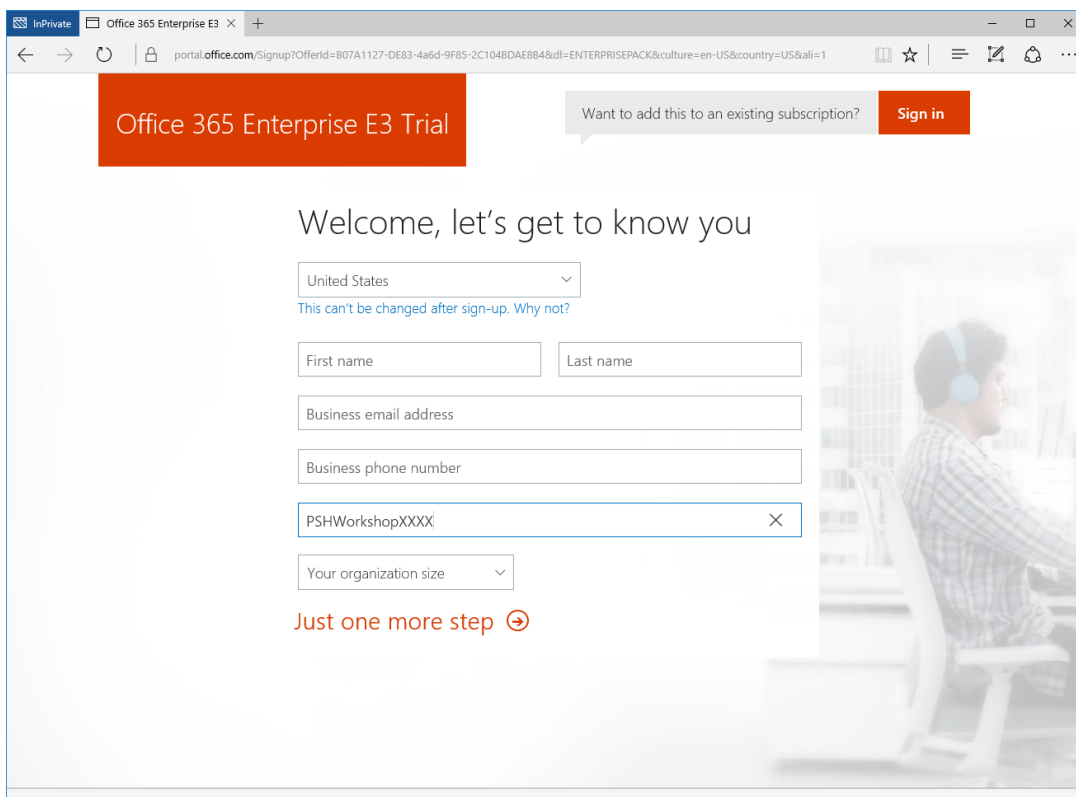
To do so, navigate to the following URL: <http://bit.ly/CompareAllO365BusinessPlans> and you'll see the following page:

| Select a plan  | Office 365 ProPlus  | Office 365 Enterprise E1                                    | Office 365 Enterprise E3  | Office 365 Enterprise E5   |
|--|---|---|---|--|
| See options for:<br>Small Business<br>Education<br>Government<br>Nonprofit<br>Home | \$12.00<br>user/month<br>(annual commitment)                      | \$8.00<br>user/month<br>(annual commitment)                 | \$20.00<br>user/month<br>(annual commitment)  | \$35.00<br>user/month<br>(annual commitment)   |
|  | <a href="#">Buy now</a>   | <a href="#">Buy now</a>                                     | <a href="#">Buy now</a>   | <a href="#">Buy now</a>  |
|  | <a href="#">Learn more</a>  | <a href="#">Learn more</a>                                  | <a href="#">Learn more</a>  | <a href="#">Learn more</a>   |
|  | Fully installed Office on PC/Mac with apps for tablets and phones | Online versions of Office with email and video conferencing | All the features of ProPlus and E1 plus compliance tools, information protection, and voicemail integration | All the features of E3 plus a new class of unified communications solutions with advanced Skype for Business meetings and voice capabilities |

Scroll down until you reach the **free trial** options for **Office 365 Enterprise E3** and **Office 365 Enterprise E5**. In our workshop we will focus on the E3 plan, so select the **free trial** option in the E3 column.

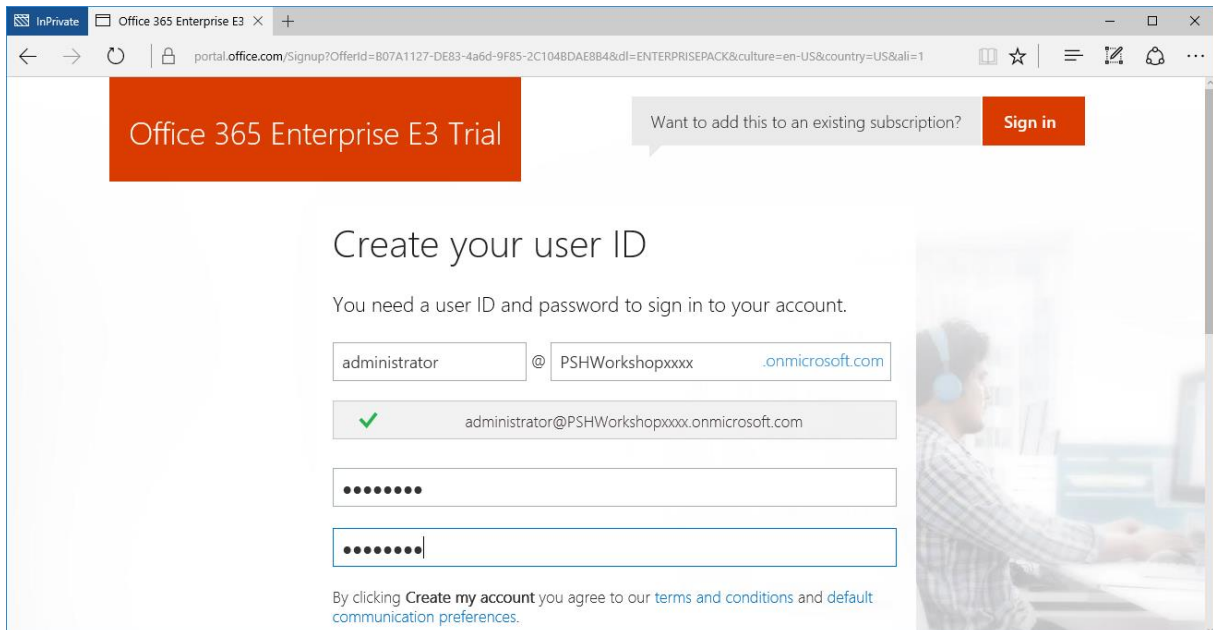


On the Welcome page, fill in the appropriate details, for the company name, use PSHWorkshopXXXX, where XXXX is a random number.



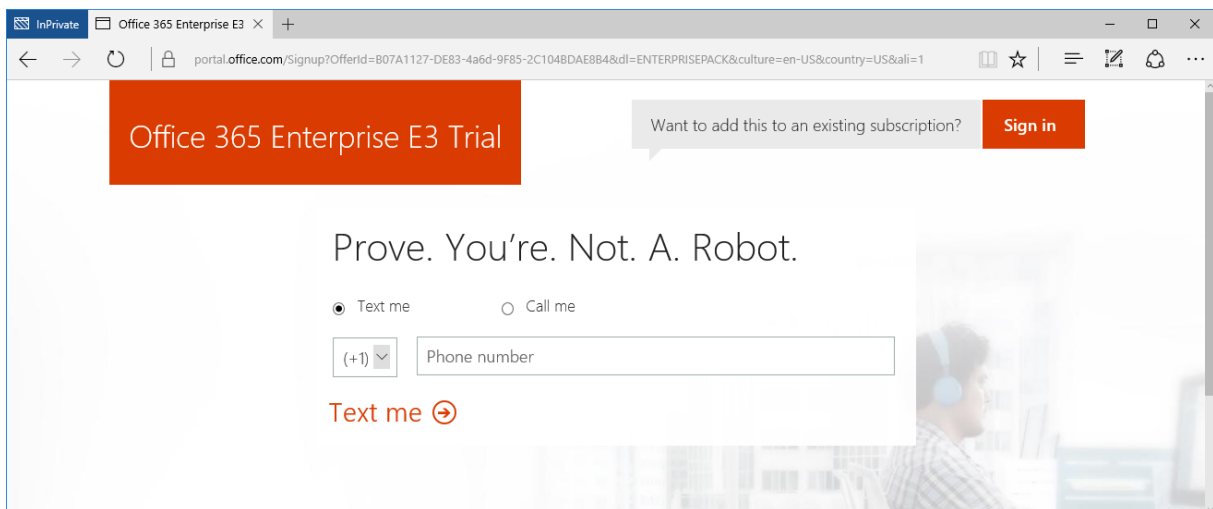
When you've entered the requested information click on **Just on more step**.

On the next page you can create your administrator user ID. Fill in a username (i.e. administrator) and company name as used in the previous page (PSHWorkshopXXXX, where XXXX is the random number you've used). Use a secure password (please do not use something like P@ssw0rd, this is too easy to guess) and click on **Create my account** (not visible in screenshot below):



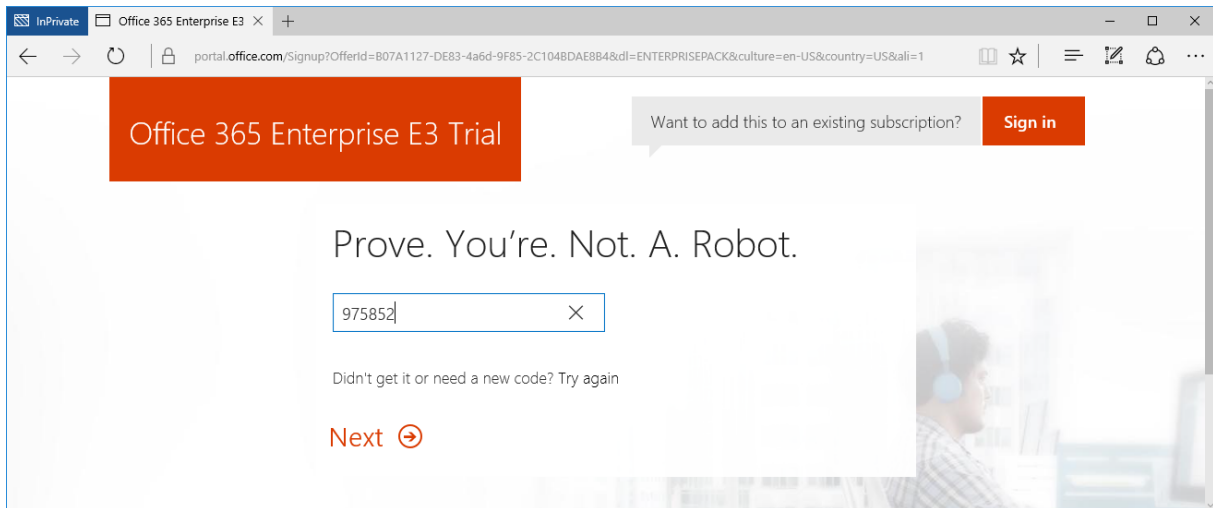
The screenshot shows a web browser window with the URL `portal.office.com/Signup?OfferId=B07A1127-DE83-4a6d-9F85-2C104BDAE8B4&dl=ENTERPRISEPACK&culture=en-US&country=US&ali=1`. The page features a red header with the text "Office 365 Enterprise E3 Trial" and a "Sign in" button. The main heading is "Create your user ID". Below it, a message states: "You need a user ID and password to sign in to your account." The form contains three input fields: a username field with "administrator", a company name field with "PSHWorkshopxxxx.onmicrosoft.com", and a password field with masked characters. A green checkmark is visible next to the company name field, and the email address "administrator@PSHWorkshopxxxx.onmicrosoft.com" is displayed below. At the bottom, there is a link to "terms and conditions and default communication preferences".

Microsoft wants to make sure you're not a robot creating fake tenants so enter your phone number in the next Windows, select **Text me** or **Call me** and click the **Text me** (or **Call me**) button:

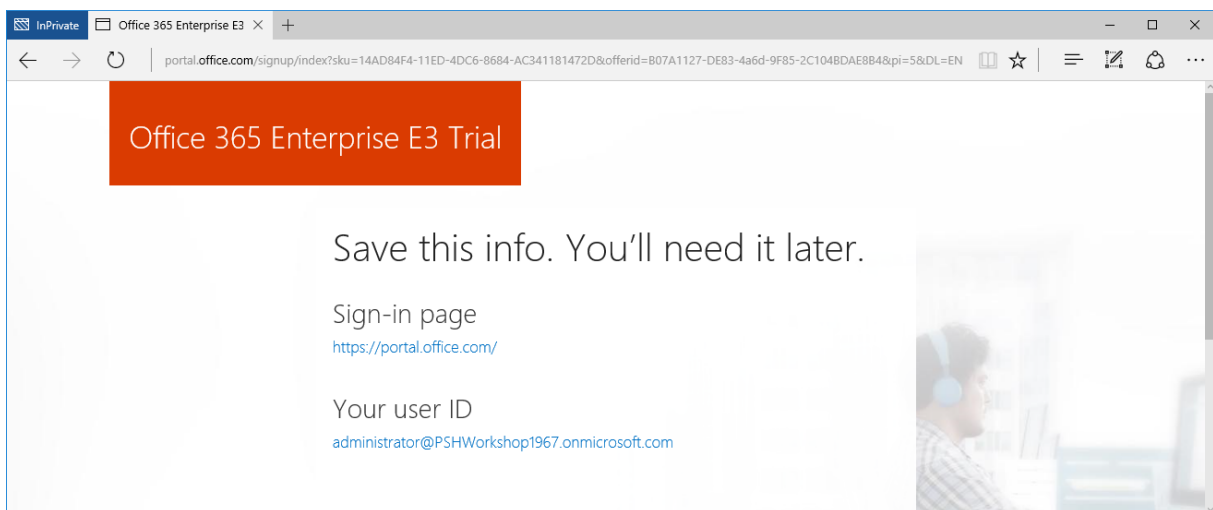


The screenshot shows the same web browser window as the previous one, but the page has advanced to the "Prove. You're. Not. A. Robot." step. The heading is "Prove. You're. Not. A. Robot." Below it, there are two radio buttons: "Text me" (selected) and "Call me". A phone number input field is present, with a dropdown menu showing "(+1)" and a "Phone number" label. A red "Text me" button with a right-pointing arrow is visible below the input field.

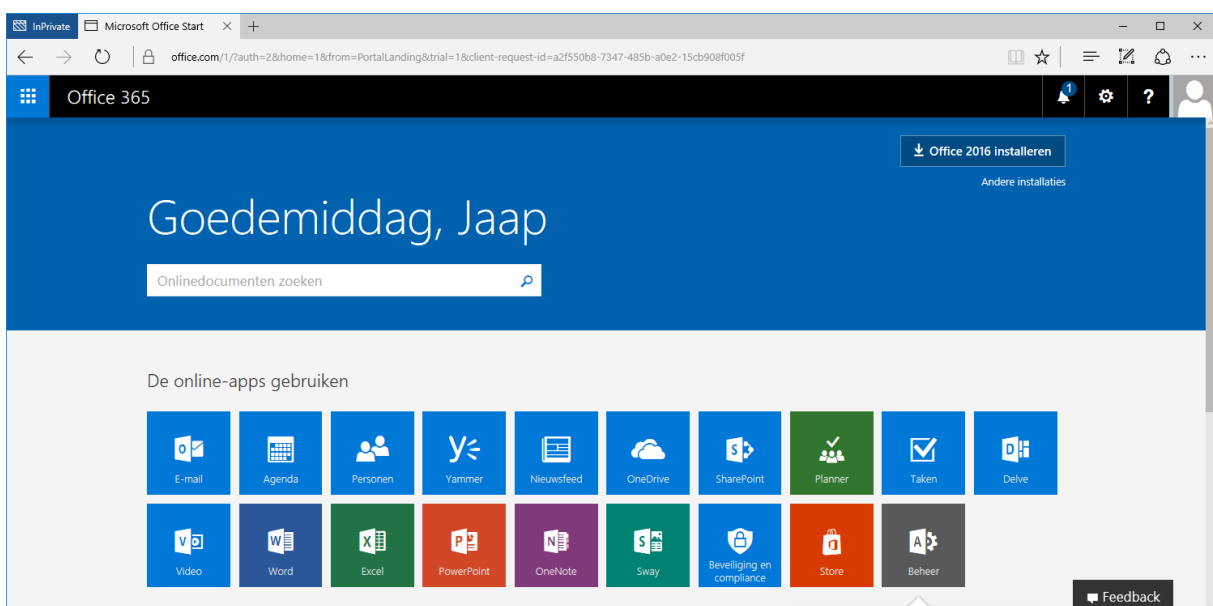
Enter the verification code you've received and click Next:



Your tenant will be created and you'll see something like the following confirmation page:



After a couple of minutes, you'll be able to logon to the Office 365 portal:



You're all set now!

## PowerShell Modules

As explained during the first session all application have a different approach when it comes to PowerShell. Exchange is relatively easy since you can connect from a regular PowerShell windows, but Azure Active Directory, Skype for Business Online and SharePoint Online have an installable PowerShell module.

Please install the following modules and see if they work in your PSHWorkshopXXXX tenant.

### Azure Active Directory

Install the Online Services Sign-in Assistant, this be downloaded from:

<http://www.microsoft.com/en-us/download/details.aspx?id=41950>

Install Azure Active Directory PowerShell module, this can be downloaded from:

<http://go.microsoft.com/fwlink/p/?linkid=236297>

More information regarding the Azure Active Directory PowerShell Module can be found on the Microsoft MSDN site (as all information can be found on MSDN):

[https://msdn.microsoft.com/en-us/library/azure/jj151815\(v=azure.98\).aspx](https://msdn.microsoft.com/en-us/library/azure/jj151815(v=azure.98).aspx)

Start the Azure Active Directory PowerShell module and enter the following commands to access you Azure AD environment. Replace the XXXX with your unique ID:

```
$Cred = Get-Credential administrator@pshworkshopxxxx.onmicrosoft.com
Connect-MsolService -Credential $Cred
```

### Exchange Online

For Exchange Online there's no need to install any separate PowerShell module. When using implicit remoting you are actually downloading all available PowerShell commands into the current session. When connecting this way the RBAC roles are preserved and you'll only have access to the cmdlets you are allowed to use.

To access Exchange Online, open a regular PowerShell window and enter the following commands:

```
$ExCred = Get-Credential administrator@pshworkshopxxxx.onmicrosoft.com
$Session = New-PSSession -ConfigurationName Microsoft.Exchange
-ConnectionUri https://ps.outlook.com/powershell/ -Credential $ExCred
-Authentication Basic -AllowRedirection
Import-PSSession $Session
```

### Skype for Business Online

For Skype for Business Online you first have to download and install the Skype for Business Online Connector Module which can be downloaded here: <http://go.microsoft.com/fwlink/?LinkId=294688>

Once installed you can import the Skype for Business Online PowerShell module using the Import-Module command:

```
Import-Module LyncOnlineConnector
```

A warning message may appear that WinRM is not running, you can click [Y] to accept starting WinRM.

To connect your Skype for Business Online PowerShell module to Skype for Business Online, you can use the following commands:

```
$credential = Get-Credential Administrator@pshworkshopXXXX.onmicrosoft.com
$session = New-CsOnlineSession -Credential $credential
Import-PSSession $session
```

You can now use all Skype for Business Online commandlets against your tenant.

**Note.** To get a listing of all available commands in Skype for Business Online you can use the following command:

```
Get-Command *CsOnline*
```

## SharePoint Online

For SharePoint Online you need to download and install a separate PowerShell module which can be found here: <https://www.microsoft.com/en-us/download/details.aspx?id=35588>

Once finished you get an additional PowerShell module on your computer which you can start directly, just like the Azure AD PowerShell module.

When you want to connect to SharePoint Online you need to know your tenant admin credentials (duh), but you also need to use your full (admin) email address and the FQDN of your SharePoint Online environment. So, if your tenant name is "PSHWorkshopXXXX" and the admin UPN (Email address) is administrator@pshworkshopXXXX.onmicrosoft.com, you can use the following commands:

```
$adminUPN="administrator@pshworkshopXXXX.onmicrosoft.com"
$TenantName=" pshworkshopXXXX"
$userCredential = Get-Credential -UserName $adminUPN
Connect-SPOService -Url https://$TenantName-admin.sharepoint.com -
Credential $userCredential
```

Again, to retrieve a list of SharePoint Online commandlets you can use the following command:

```
Get-Command *SPO*
```

## Microsoft Azure Active Directory

In this hands-on part you will be creating a new domain in Azure Active Directory, that is, when you have access to a regular test domain (that's owned by you).

### Create a new Domain

Use the `New-MsolDomain` command to create a new domain in Azure Active Directory. Use the `Get-MsolDomainVerificationDns` and the `Confirm-MsolDomain` to verify and confirm the domain registration.

### Licenses

Use the `Get-MsolUser` command to retrieve a list of unlicensed users in your organization. How many users do you see and how many are unlicensed?

Use the `Get-MsolAccountSKU` command to view the subscription information in your tenant. How many licenses are issued?

### User Management

Use the `New-MsolUser` command to create 4 different users:

```
New-MsolUser -UserPrincipalName
Santa@pshworkshopXXXX.onmicrosoft.com -FirstName Santa -LastName
Klaus -DisplayName 'Santa Klaus' -Password 'Pass2016' -
ForceChangePassword:$TRUE -LicenseAssignment
"pshworkshopXXXX:ENTERPRISEPACK" -UsageLocation NL

New-MsolUser -UserPrincipalName
OtherSanta@pshworkshopXXXX.onmicrosoft.com -FirstName Other -
LastName Santa -DisplayName 'Other Santa Klaus' -LicenseAssignment
"pshworkshopXXXX:ENTERPRISEPACK" -UsageLocation NL

New-MsolUser -UserPrincipalName
YetAnother@pshworkshopXXXX.onmicrosoft.com -FirstName Yet -LastName
Another -DisplayName 'Yet another Santa Klaus'

New-MsolUser -UserPrincipalName John@pshworkshopXXXX.onmicrosoft.com
-FirstName John -LastName Brown -DisplayName 'John Brown'
```

Use the `Get-MsolUser` command again to retrieve a list of user accounts and identify which users do not have a license issued. Use the `Set-MsolUserLicense` command to issue licenses to the unlicensed users.

If you want, use the `New-MsolLicenseOptions` command to create an individual license plan and issue this license to a user that you've created.

Use the `Set-MsolUserPassword` command to change the password for user John Smith. For the domain you just created use the `Set-MsolPasswordPolicy` command to set the password to 'never expire'.

## Bulk Management

Use the `Import-CSV` command to import the users listed in the bulk-import.csv file into your organization. The CSV file is listed at the end of this document. Copy and paste the list in Notepad and store it as c:\temp\bulk-import.csv and c:\temp\delete-users.csv

## Group Management

Use the `New-MsolGroup` to create a security group called "Marketing". Please note the GroupObjectID since it is used in subsequent commands.

Use the `Add-MsolGroupMember` command to add user john@pshworkshopXXXX.com to the Marketing security group you just created.

When done, check the security group using the Azure Portal. Delete the security group using the `Remove-MsolGroup` command.

## Exchange Online

## SharePoint Online

Connect to the SPO environment using the following command:

```
$adminUPN="administrator@pshworkshopXXXX.onmicrosoft.com"
$TenantName=" pshworkshopXXXX"
$userCredential = Get-Credential -UserName $adminUPN
Connect-SPOService -Url https://$TenantName-admin.sharepoint.com -
Credential $userCredential
```

In SharePoint Online it can be very useful to work with variables. Just put a URL in a variable, for example:

```
$Site = https://pshworkshopXXXX.sharepoint.com
```

Now you can use a construction like this:

```
Get-SPOSite -Identity $Site
```

For a sales and a marketing site you can use the following variables:

```
$Sales = https://pshworkshopXXXX.sharepoint.com/sites/sales
$Marketing = https://pshworkshopXXXX.sharepoint.com/sites/marketing
```

Use the `Get-SPOSite` command in a PowerShell window to retrieve a list of sites. Compare this to the view you have when logging on into the SharePoint portal https://pshworkshopXXXX-admin.sharepoint.com.



Use the `Get-SPOWebTemplate` command to retrieve a list of templates you can use when creating new sites.

Use the `New-SPOSite` command to create a new site called "sales site" and add user John to the site using the `Add-SPOUser` command. The user should be added to "Team Site Members"

Repeat this step to create a site called "marketing site" and add user John to this site.

When finished try to login with John's user account to the Sales or Marketing site. Make sure the previous commands were finished successfully.

When done use the `Remove-SPOSite` command to delete the Sales and Marketing sites. Wait a minute and try to restore the Sales site. When restored the Sales site check its existence in the SharePoint Online portal <https://pshworkshopXXXX-admin.sharepoint.com>.

## Skype for Business Online

```
$Credential = Get-Credential
$session = New-CsOnlineSession -Credential $credential
Import-PSSession $session
```

```
Set-CSPrivacyConfiguration -EnablePrivacyMode $True
Set-CSPushNotificationConfiguration -EnableApplePushNotification $False
Get-CSPrivacyConfiguration
Get-CSPushNotificationConfiguration
Set-CsTenantFederationConfiguration -AllowPublicUsers $True
Set-CsTenantFederationConfiguration -AllowFederatedUsers $True
$x = New-CsEdgeDomainPattern -Domain "litware.com"
$newAllowList = New-CsEdgeAllowList -AllowedDomain $x
Set-CsTenantFederationConfiguration -AllowedDomains $newAllowList
Get-CsTenantFederationConfiguration
Remove-PSSession $session
```

## PowerShell commands

In the following sections you will find some PowerShell commands that are used in the various topics.

### Azure Active Directory

#### Domain Management

```
New-MsolDomain -Authentication Managed -Name pshworkshopXXXX.com  
Get-MsolDomainVerificationDns -DomainName pshworkshopXXXX.com -Mode  
DnsTxtRecord  
Confirm-MsolDomain -DomainName pshworkshopXXXX.com
```

```
Get-MsolDomain -Identity pshworkshopXXXX.com | select Name,Capabilities
```

#### License Management

```
Get-MsolAccountSku | Where {$_.SkuPartNumber -eq "ENTERPRISEPACK"} |  
ForEach {$_.ServiceStatus}  
  
$Options = New-MsolLicenseOptions -AccountSkuId  
pshworkshopXXXX:ENTERPRISEPACK -DisabledPlans  
YAMMER_ENTERPRISE,RMS_S_ENTERPRISE,SHAREPOINTWAC,SHAREPOINTENTERPRISE  
  
Set-MsolUserLicense -UserPrincipalName john@pshworkshopXXXX.onmicrosoft.com  
-LicenseOptions $Options
```

```
Get-MsolUser -UnlicensedUsersOnly  
  
Get-MsolUser -All | ft displayname,Licenses | Out-File  
"c:\userlicenses.csv"  
  
Set-MsolUserLicense -UserPrincipalName john@pshworkshopXXXX.onmicrosoft.com  
-AddLicenses "pshworkshopXXXX:ENTERPRISEPACK"  
  
Set-MsolUserLicense -UserPrincipalName john@pshworkshopXXXX.onmicrosoft.com  
-RemoveLicenses "pshworkshopXXXX:ENTERPRISEPACK"
```

#### User Management

```
New-MsolUser -UserPrincipalName Santa@pshworkshopXXXX.onmicrosoft.com -  
FirstName Santa -LastName Klaus -DisplayName 'Santa Klaus' -Password
```

```
'Pass2016' -ForceChangePassword:$TRUE -LicenseAssignment
"pshworkshopXXXX:ENTERPRISEPACK" -UsageLocation NL

New-MsolUser -UserPrincipalName OtherSanta@pshworkshopXXXX.onmicrosoft.com
-FirstName Other -LastName Santa -DisplayName 'Other Santa Klaus' -
LicenseAssignment "pshworkshopXXXX:ENTERPRISEPACK" -UsageLocation NL

New-MsolUser -UserPrincipalName YetAnother@pshworkshopXXXX.onmicrosoft.com
-FirstName Yet -LastName Another -DisplayName 'Yet another Santa Klaus'
```

```
Set-MsolUser -UserPrincipalName john@pshworkshopXXXX.onmicrosofft.com -
BlockCredential $true

Remove-MsolUser -UserPrincipalName john@pshworkshopXXXX.onmicrosofft.com -
Force

Get-MsolUser -ReturnDeletedUsers

Restore-MsolUser -UserPrincipalName john@ppshworkshopXXXX.onmicrosofft.com

Get-MsolUser -ReturnDeletedUsers | Remove-MsolUser -RemoveFromRecycleBin -
Force
```

```
Set-MsolUserPassword -UserPrincipalName
john@pshworkshopXXXX.onmicrosoft.com -NewPassword "Pass2016" -
ForceChangePassword $TRUE

Set-MsolUser -UserPrincipalName john@pshworkshopXXXX.onmicrosoft.com -
PasswordNeverExpires $true

Set-MsolPasswordPolicy -ValidityPeriod 60 -NotificationDays 14 -DomainName
pshworkshopXXXX.onmicrosoft.com
```

## Bulk User Management

```
Import-Csv -Path c:\temp\bulk-import.csv | ForEach-Object {New-MsolUser -
FirstName $_.FirstName -LastName $_.LastName -UserPrincipalName
$_.UserPrincipalName -DisplayName "$($_.FirstName) $($_.LastName)" -
LicenseAssignment "pshworkshopXXXX:ENTERPRISEPACK" -UsageLocation NL}

Import-Csv -Path C:\temp\delete-users.csv | ForEach-Object {Remove-MsolUser
-UserPrincipalName $_.UserPrincipalName -Force}
```

The CSV file contains the following:

```
FirstName,LastName,UserPrincipalName
James,Bond,James@pshworkshopXXXX.onmicrosoft.com
Miss,MoneyPenney,miss@pshworkshopXXXX.onmicrosoft.com
Jaap,Wesselius,jaap@pshworkshopXXXX.onmicrosoft.com
```

```
Michel,DeRooij,michel@pshworkshopXXXX.onmicrosoft.com
Jack,Smith,jack@pshworkshopXXXX.onmicrosoft.com
Tony,Redmond,tony@pshworkshopXXXX.onmicrosoft.com
Beau,Terham,beau@pshworkshopXXXX.onmicrosoft.com
Duw,Vandenberg,duw@pshworkshopXXXX.onmicrosoft.com
Blake,Edwards,blake@pshworkshopXXXX.onmicrosoft.com
Blake,Thompson,blakeT@pshworkshopXXXX.onmicrosoft.com
Sun,Chaney,sun@pshworkshopXXXX.onmicrosoft.com
John,Olovski,olovski@pshworkshopXXXX.onmicrosoft.com
```

Store this information in c:\temp as bulk-import.csv and delete-users.csv. Don't forget to include the first line.

## Group Management

```
New-MsolGroup -DisplayName "Sales" -Description "Sales Team"
```

GroupobjectID = 7210a7de-96b0-4307-b66d-4d632507ceff (**different per group/tenant/time etc**)

```
Get-MsolUser -UserPrincipalName John@pshworkshopXXXX.onmicrosoft.com |
Select UserPrincipalName,ObjectId

Add-MsolGroupMember -GroupmemberObjectID 372a3958-a482-4eea-abfe-
808549e3b93f -GroupobjectID 7210a7de-96b0-4307-b66d-4d632507ceff
```

```
$UserID = Get-MsolUser -SearchString "john"
$groupId = Get-MsolGroup -searchString "Sales"
Add-MsolGroupMember -GroupmemberObjectID $UserID.ObjectID -GroupobjectID
$groupId.ObjectID
```

```
$UserID = Get-MsolUser -SearchString "Santa Klaus"
$groupId = Get-MsolGroup -searchString "Sales"
Add-MsolGroupMember -GroupmemberObjectID $UserID.ObjectID -GroupobjectID
$groupId.ObjectID
```

```
Remove-MsolGroupMember -GroupmemberObjectID $UserID.ObjectID -GroupobjectID
$groupId.ObjectID
```

```
Remove-MsolGroup -ObjectID 31650ad4-5a14-4272-a83f-f04b5a4f71a8 -Force
```

```
$groupId = Get-MsolGroup -searchString "Sales"  
Remove-MsolGroup -objectid $groupId.ObjectId
```

## SharePoint Online

```
$Cred = Get-Credential administrator@pshworkshopXXXX.onmicrosoft.com  
$TenantName = "pshworkshopXXXX"  
$Site = "https://pshworkshopXXXX.sharepoint.com"  
Connect-SPOService -Url https://$TenantName-admin.sharepoint.com -  
Credential $Cred
```

```
Get-SPOSite  
Get-SPOSite -Identity https://pshworkshopXXXX.onmicrosoft.com  
Get-SPOSite -Identity $Site  
Get-SPOWebTemplate
```

```
Get-SPOUser -Site https://pshworkshopXXXX.sharepoint.com  
Get-SPOUser -Site $Site
```

```
Get-SPOSiteGroup -Site https://pshworkshopXXXX.sharepoint.com  
Get-SPOSiteGroup -Site $Site
```

```
New-SPOSite -Url https://pshworkshopXXXX.sharepoint.com/sites/marketing -  
Owner administrator@pshworkshopXXXX.onmicrosoft.com -StorageQuota 500 -  
Title "Marketing Site" -Template STS#0  
  
Add-SPOUser -Site https://pshworkshopXXXX.sharepoint.com -LoginName  
santa@pshworkshopXXXX.onmicrosoft.com -Group "Team Site Members"  
  
Add-SPOUser -Site $marketing -LoginName  
santa@pshworkshopXXXX.onmicrosoft.com -Group "Team Site Members"
```

```
Set-SPOSite -Url https://pshworkshopXXXX.sharepoint.com/sites/sales -  
StorageQuota 1000 -StorageQuotaWarningLevel 750
```

```
Remove-SPOSite -Identity https://pshworkshopXXXX.sharepoint.com/sites/sales  
-NoWait
```

```
Restore-SPODeletedSite -Identity  
https://pshworkshopXXXX.sharepoint.com/sites/sales -NoWait
```