Office365, Exchange, SharePoint and Skype For Business Powershell Profile Script

Hi Everyone.

This Script creates powershell functions to be used at your profile to make your life ease when administering Office365, Exchange, Azure AD, SharePoint and Skype For Business Powershell.

########## **Setup your powershell profile** ###########

**Open powershell as administrator**

#1 list all user's powershell profile

**$profile | Get-Member -MemberType NoteProperty | fl name,definition <enter>**

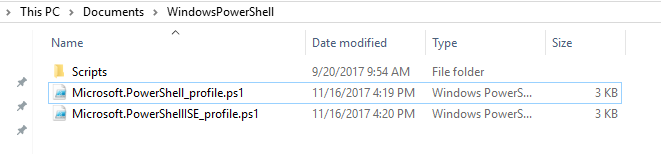
#2 test your profile path / file

**Test-Path $profile <enter> ; if it returns false create a new profile**

#3 create a new profile item (run powershell ISE and powershell as administrator and repeat that for each powershell)

**New-Item $profile -ItemType File -Force <enter>**

You'll have the following



#4 edit your profile file and copy and paste the powershell code attached.

**powershell\_ise.exe $profile <enter>**

#5 install all modules (Windows 10, Server 2012+)

Install-Module MicrosoftTeams

Install-Module MSOnline

Install-Module AzureAD

Install-Module AzureRM

or use **-Force t**o upgrade all modules

Install-Module MicrosoftTeams -Force

Install-Module MSOnline -Force

Install-Module AzureAD -Force

Install-Module AzureRM -Force

########## **Functions**###########

Function **Connect-Office365Full** = connect to all Enabled services at once

Function **Disconnect-Office365** = disconnect all sessions opened

Function **Change-Office365Full**= disconnect all sessions / remove and then asks you the credentials for new **Domain**to be managed\*.

Function **O365-Export-Guest-Users**: export all guest users on a tenant (save on the current directory)

Function **O365-Export-MFA-Disabled-Users**: export all users with MFA disabled

Function **O365-Export-MFA-Disabled-Users-OnScreen**: export all users with MFA disabled on screen

Function **Connect-Tenant**= connects at Skype For Business Online

Function **Connect-SharePointOnline** = connects at Sharepoint Online

\*[We have noticed that when we manage a lot of domain at the same we can experience issues]

########## **Modules download links** ###########

Download and install all required modules at you workstation / server

Windows Azure <http://go.microsoft.com/fwlink/p/?LinkId=286152> , <http://go.microsoft.com/fwlink/p/?linkid=236297>

Skype For Business <https://www.microsoft.com/en-us/download/confirmation.aspx?id=39366>

Sharepoint <https://www.microsoft.com/en-us/download/details.aspx?id=35588>

**PowerShell**

################################################################################

#Author Thiago Beier thiago.beier@gmail.com

#Version: 2.1 on April 30, 2019

#Toronto,CANADA

#Powershell Functions To Manager Office 365, Exchange Online, SharePoint and Skype for Business.

#Before using this please check the PowerShell modules required

########## Modules download links ##########

#Windows Azure http://go.microsoft.com/fwlink/p/?LinkId=286152 , http://go.microsoft.com/fwlink/p/?linkid=236297

#Skype For Business https://www.microsoft.com/en-us/download/confirmation.aspx?id=39366

#Sharepoint https://www.microsoft.com/en-us/download/details.aspx?id=35588

#BEGIN

**Function** Connect-Office365Full {

    #exchange

    $URL = "https://ps.outlook.com/powershell"

    $Credentials = **Get-Credential** -Message "Enter your Office 365 admin credentials"

    $EXOSession = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionUri $URL -Credential $Credentials -Authentication Basic -AllowRedirection -Name "ExchangeOnline"

    Import-PSSession $EXOSession

    #office365

    Import-Module MsOnline

    Connect-MsolService -Credential $Credentials

    #skype for business

    Import-Module LyncOnlineConnector

    $sfbSession = New-CsOnlineSession -Credential $Credentials

    Import-PSSession $sfbSession

    #azure AD

    Connect-AzureAD -Credential $Credentials -Confirm:$false

}

**Function** Connect-Tenant {

    #exchange tenant

    $domainname = **Read-Host** -Prompt "Enter your tenant domain"

    $tenantadminname = **Read-Host** -Prompt "Enter your tenant admin account name"

    $AdminName = "$tenantadminname@$domainname"

**Read-Host** -Prompt "Enter your tenant password" -AsSecureString | **ConvertFrom-SecureString** | Out-File cred.txt

    $Pass = **Get-Content** cred.txt | **ConvertTo-SecureString**

    $cred = **new-object** -typename System.Management.Automation.PSCredential -argumentlist $AdminName, $Pass

    Import-Module MSOnline

    Connect-MsolService -Credential $cred

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

    Import-PSSession $EXOSession -AllowClobber

    Connect-AzureAD -Credential $cred -Confirm:$false

}

**function** Get-O365GuestsUsersThroughAzureAD

{

**param**($sCSVFileName)

**Try**

    {

        [array]$O365GuestsUsers = $null

        $O365TenantGuestsUsers=Get-AzureADUser -**Filter** "Usertype eq 'Guest'”

**foreach** ($O365GuestUser **in** $O365TenantGuestsUsers)

        {

            $O365GuestsUsers=**New-Object** PSObject

            $O365GuestsUsers | **Add-Member** NoteProperty -Name "Guest User DisplayName" -Value $O365GuestUser.DisplayName

            $O365GuestsUsers | **Add-Member** NoteProperty -Name "User Principal Name" -Value $O365GuestUser.UserPrincipalName

            $O365GuestsUsers | **Add-Member** NoteProperty -Name "Mail Address" -Value $O365GuestUser.Mail

            $O365AllGuestsUsers+=$O365GuestsUsers

        }

        $O365AllGuestsUsers | **Export-Csv** $sCSVFileName

    }

**catch** [System.Exception]

    {

        Write-Host -ForegroundColor Red $\_.Exception.ToString()

    }

}

**Function** O365-Export-Guest-Users {

    $sCSVFileName="AllTenantGuestsUsers-$(get-date -f yyyy-MM-dd-HH-mm-ss).csv"

    #Getting Tenant Guests Users

    Get-O365GuestsUsersThroughAzureAD -sCSVFileName $sCSVFileName

}

**Function** O365-Export-MFA-Disabled-Users {

$allmfadisabledusers = Get-MsolUser -All | where {$\_.isLicensed -eq $true} | **select** DisplayName,UserPrincipalName,@{N="MFA Status"; E={ **if**( $\_.StrongAuthenticationRequirements.State -ne $null){ $\_.StrongAuthenticationRequirements.State} **else** { "Disabled"}}}

$allmfadisabledusers

$totalmfadisabledusers = ($allmfadisabledusers).count

write-host -ForegroundColor Yellow "You have $totalmfadisabledusers Users without MFA enabled"

}

**Function** O365-Export-MFA-Disabled-Users-OnScreen {

Get-MsolUser -All | where {$\_.isLicensed -eq $true} | **select** DisplayName,UserPrincipalName,@{N="MFA Status"; E={ **if**( $\_.StrongAuthenticationRequirements.State -ne $null){ $\_.StrongAuthenticationRequirements.State} **else** { "Disabled"}}} | Out-GridView

}

**Function** Disconnect-Office365Full {

    Get-PSSession | Remove-PSSession

}

**Function** Change-Office365Full {

    Disconnect-Office365Full

    Connect-Office365Full

}

**Function** Connect-SharepointOnline {

$adminUPN=**read-host** "<the full email address of a SharePoint Online global administrator account, example: jdoe@contosotoycompany.onmicrosoft.com>"

$orgName=**read-host** "<name of your Office 365 organization, example: contosotoycompany>"

Connect-SPOService -Url https://$orgName-admin.sharepoint.com -Credential $Credentials

}

#END