Office365 & Azure Powershell Management Functions

This Powershell profile comes with the following Functions:

* **Connect-Office365Full** : connect to a Office 365 tenant / subscription
* **Change-Office365Full** : change from a Office 365 tenant / subscription into another (avoid mistakes when managing several tenants at the same time)
* **Disconnect-Office365Full** : disconnect from a Office 365 tenant / subscription (close all powershell sessions opened)
* **Connect-Tenant** : connect to a specific and standardized Office 365 tenant / Azure subscription
* **O365-Export-Guest-Users** : this functoin exports to CSV at the current logged directory when powershell is loaded all Guest's users information
* **O365-Export-MFA-Disabled-Users** : exports to screen all users with MFA disabled
* **O365-Export-MFA-Disabled-Users-OnScreen** : exports to a powershell grid-view all users with MFA disabled
* **Connect-SharepointOnline** : this function only calls and connect to a Office365 Sharepoint Administration

The Connect-Tenant funcition we use to standardize our Office365 or Azure Tenant Management once we have a starndardized support user on each managed tenant.

Those funcionts are well used on my daily tasks.

**PowerShell**

#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"

    $AdminName = "supportuser@$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

Follow the instructions from my previous post <https://gallery.technet.microsoft.com/scriptcenter/Office365-Exchange-bb504cce?redir=0> to set up by the 1st time or to update your Powershell Script under your user account.

**References**

Powershell functions <https://en.wikiversity.org/wiki/PowerShell/Functions> ; <https://en.wikiversity.org/wiki/PowerShell/Functions>

Exchange On-line functions <https://docs.microsoft.com/en-us/powershell/exchange/exchange-online/connect-to-exchange-online-powershell/connect-to-exchange-online-powershell?view=exchange-ps>