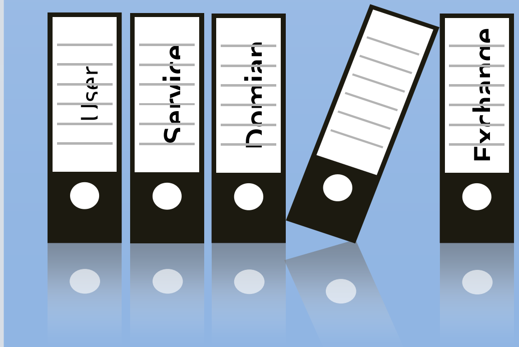
Windows PowerShell cmdlets

For

Office 365

Microsoft Office Professional Plus
Microsoft Exchange Online
Microsoft SharePoint Online
Microsoft Lync Online
Microsoft Office 365



Microsoft Certificates:

- ❖ Introduction to Selling Microsoft Online Services to Partners
- ❖ MPN Technical assessment for Microsoft Office 365
- ❖ MPN Sales and Service Assessment for Microsoft Online Services

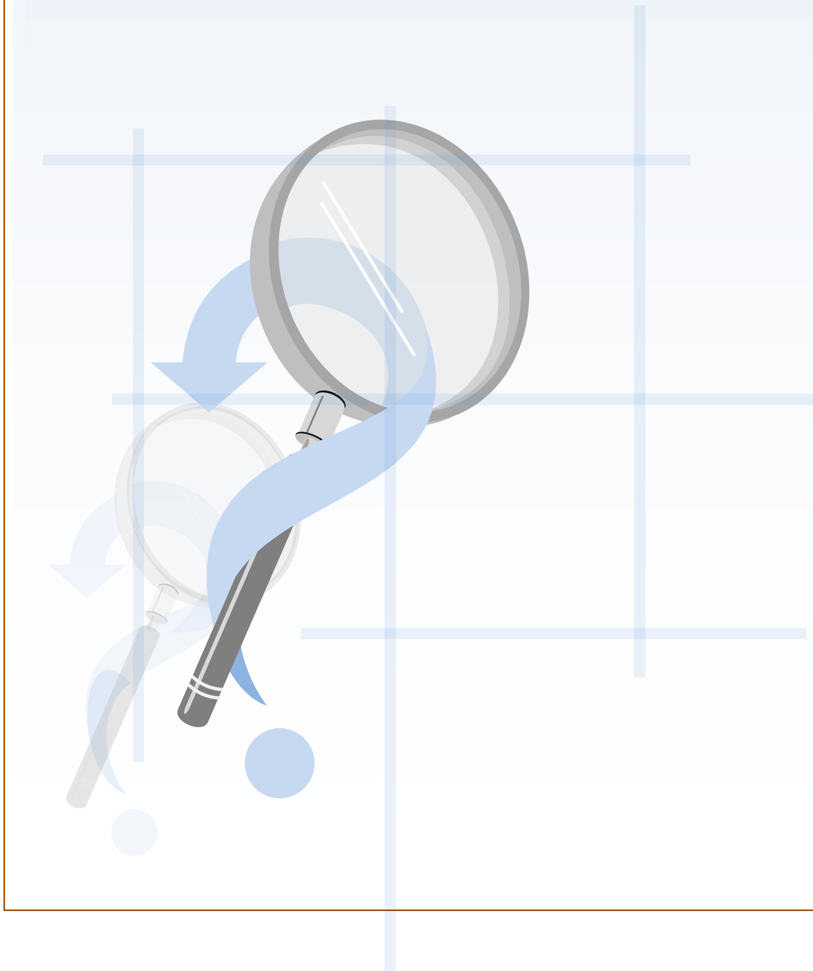
Muthu Swamy S, APSM, PMP, CSM, PM-2008
Friday, October 1, 2010
Rms.reddy@hotmail.com

Guideline

This article is explaining the step by step guidelines to manage Users, Groups, Service Principals, Domain, Single Sign on, Subscription and Exchange using Windows PowerShell cmdlets for Office 365.

The targeted audience is an Administrator of Office 365 Online Services.

Table of Contents



Step-1: Users

Windows PowerShell cmdlet	Description
Convert-MsolFederatedUser	The Convert-MsolFederatedUser cmdlet is used to update a user in a domain that was recently converted from single sign-on (also known as identity federation) to standard authentication type. A new password must be provided for the user.
Get-MsolUser	The Get-MsolUser cmdlet can be used to retrieve an individual user, or list of users. An individual user will be retrieved if the ObjectId or UserPrincipalName parameter is used.
New-MsolUser	The New-MsolUser cmdlet is used to create a new user in the Microsoft Online directory. In order to give the user access to services, they must also be assigned a license (using the LicenseAssignment parameter).
Remove-MsolUser	The Remove-MsolUser cmdlet is used to remove a user from the Microsoft Online directory. This cmdlet will delete the user, their licenses, and any other associated data.
Restore-MsolUser	The Restore-MsolUser cmdlet restores a user that is in the Deleted users view to their original state. Users will remain in the Deleted users view for 30 days.
Set-MsolUser	The Set-MsolUser cmdlet is used to update a user object. Note that this cmdlet should be used for basic properties only. The licenses, password, and User Principal Name for a user can be updated through Set-MsolUserLicense, Set-MsolUserPassword and Set-MsolUserPrincipalName cmdlets respectively.
Set-MsolUserPassword	The Set-MsolUserPassword cmdlet is used to change the password of a user. This cmdlet can only be used for users with standard identities.
Set-MsolUserPrincipalName	The Set-MsolUserPrincipalName cmdlet is used to change the User Principal Name (user ID) of a user. This cmdlet can be used to move a user between a federated and standard domain, which will result in their authentication type changing to that of the target domain.
Set-MsolPasswordPolicy	The Set-MsolPasswordPolicy cmdlet can be used to update the password policy of a specified domain or tenant. Two settings are required, the first is to indicate the length of time that a password remains valid before it must be changed and the second is to indicate the number of days before the password expiration date that will trigger when users will receive their first notification that their password will soon expire.
Get-MsolPasswordPolicy	The Get-MsolPasswordPolicy cmdlet can be used to retrieve the values associated with the Password Expiry window or Password Expiry Notification window for a tenant or specified domain. When a domain name is specified, it must be a verified domain for the company.

Step-2: Group & Role Membership

Manage group and role membership

Use the following cmdlets to perform a variety of tasks related to group and role membership, including adding a user to a role or group, creating groups, and removing groups.

Windows PowerShell cmdlet	Description
Add-MsolGroupMember	The Add-MsolGroupMember cmdlet is used to add members to a security group. The new members can be either users or other security groups.
Get-MsolGroup	The Get-MsolGroup cmdlet is used to retrieve groups from Office 365. This cmdlet can be used to return a single group (if ObjectId is passed in), or to search within all groups.
Get-MsolGroupMember	The Get-MsolGroupMember cmdlet is used to retrieve members of the specified group. The members can be either users or groups.
New-MsolGroup	The New-MsolGroup cmdlet is used to add a new security group to the Microsoft Online directory.
Remove-MsolGroup	The Remove-MsolGroup cmdlet is used to delete a group from the Microsoft Online directory.
Remove-MsolGroupMember	The Remove-MsolGroupMember cmdlet is used to remove a member from a security group. This member can be either a user or a group.
Set-MsolGroup	The Set-MsolGroup cmdlet is used to update the properties of a security group.
Add-MsolRoleMember	The Add-MsolRoleMember cmdlet is used to add a member to a role. Currently, only users can be added to a role (adding a security group is not supported).
Get-MsolRole	The Get-MsolRole cmdlet can be used to retrieve a list of administrator roles.
Get-MsolUserRole	The Get-MsolUserRole cmdlet is used to retrieve all of the administrator roles that the specified user belongs to. This cmdlet will also return roles that the user is a member of through security group membership.
Get-MsolRoleMember	The Get-MsolRoleMember cmdlet is used to retrieve all members of the specified role.
Remove-MsolRoleMember	The Remove-MsolRoleMember cmdlet is used to remove a user from an administrator role.

Step-3: Manage Service Principals

Use the following cmdlets to perform a variety of tasks related to service principals.

Windows PowerShell cmdlet	Description
Set-MsolServicePrincipal	The Set-MsolServicePrincipal cmdlet updates a service principal in the Microsoft Online directory. It can be used to update the display name, enable/disable the service principal, trusted for delegation, the service principal names (SPNs) or the addresses.
New-MsolServicePrincipal	The New-MsolServicePrincipal cmdlet creates a service principal that can be used to represent a Line Of Business (LOB) application or an on-premises server such as Microsoft Exchange, SharePoint or Lync in the Microsoft Online directory as "service principal" objects. Adding a new application as a service principal allows that application to authenticate to other services such as Microsoft Office 365.
Get-MsolServicePrincipal	The Get-MsolServicePrincipal cmdlet can be used to retrieve a service principal or a list of service principals from the Microsoft Online directory.
Remove-MsolServicePrincipal	The Remove-MsolServicePrincipal cmdlet removes a service principal from the Microsoft Online directory.
New-MsolServicePrincipalAddress	The New-MsolServicePrincipalAddress cmdlet creates a new service principal address object that can be used to update the addresses for a service principal.
Get-MsolServicePrincipalCredential	The Get-MsolServicePrincipalCredential cmdlet can be used to retrieve a list of credentials associated with a service principal.
New-MsolServicePrincipalCredential	The New-MsolServicePrincipalCredential cmdlet can be used to add a new credential to a service principal or to add or roll credential keys for an application. The service principal is identified by supplying either the object ID, application ID, or service principal name (SPN).
Remove-MsolServicePrincipalCredential	The Remove-MsolServicePrincipalCredential cmdlet can be used to remove a credential key from a service principal in the case of a compromise or as part of credential key rollover expiration. The service principal is identified by supplying either the object ID, application ID, or service principal name (SPN). The credential to be removed is identified by its key ID.

Step-4: Manage Domains

Use the following cmdlets to perform a variety of domain management tasks, including creating or removing a domain.

Windows PowerShell cmdlet	Description
Confirm-MsolDomain	The Confirm-MsolDomain cmdlet is used to confirm ownership of a domain. In order to confirm ownership, a custom TXT DNS record must be added for the domain. The domain must first be added using the Add-MsolDomain cmdlet, and then the Get-MsolDomainVerificationDNS cmdlet should be called to retrieve the details of the DNS record that must be set.Note that there may be a delay (15 to 60 minutes) between when the DNS update is made and when the cmdlet is able to confirm ownership of a domain.
Get-MsolDomain	The Get-MsolDomain cmdlet is used to retrieve company domains.
Get-MsolDomainVerificationDns	The Get-MsolDomainVerificationDns cmdlet is used to return the DNS records that need to be set to verify a domain.
New-MsolDomain	The New-MsolDomain cmdlet is used to create a new domain object. This cmdlet can be used to create a domain with managed or federated identities, although the New-MsolFederatedDomain cmdlet should be used for federated domains in order to ensure proper setup.
Remove-MsolDomain	The Remove-MsolDomain cmdlet is used to delete a domain from the Microsoft Online directory. The domain being deleted must be empty; that is, there cannot be any users or groups with email addresses in this domain.
Set-MsolDomain	The Set-MsolDomain cmdlet is used to update settings for a domain. Using this cmdlet, the default domain can be changed, or the capabilities (Email, Sharepoint, OfficeCommunicationsOnline) can be changed.
Set-MsolDomainAuthentication	The Set-MsolDomainAuthentication cmdlet is used to change the domain authentication between standard identity and single sign-on. This cmdlet will only update the settings in Office 365; typically the Convert-MsolDomainToStandard or Convert-MsolDomainToFederated should be used instead.

Step-5: Manage Single Sign-On

Use the following cmdlets to perform tasks related to single sign-on, such as adding a new single sign-on domain (also known as identity-federated domain) to Office 365.

Windows PowerShell cmdlet	Description
New-MsolFederatedDomain	The New-MsolFederatedDomain cmdlet adds a new single sign-on domain (also known as identity-federated domain) to Office 365 and configures the relying party trust settings between the on-premises Active Directory Federation Services 2.0 server and Office 365. Due to domain verification requirements, you may need to run this cmdlet several times in order to complete the process of adding the new single sign-on domain.
Convert-MsolDomainToStandard	The Convert-MsolDomainToStandard cmdlet converts the specified domain from single sign-on (also known as identity federation) to standard authentication. This process also removes the relying party trust settings in the Active Directory Federation Services 2.0 server and Office 365. After the conversion, this cmdlet will convert all existing users from single sign-on to standard authentication. Any existing user who was configured for single sign-on will be given a new temporary password as part of the conversion process. Each converted user name and new temporary password will be recorded in a file for reference by the administrator. The administrator can then distribute the new temporary password to each converted user to enable the user to sign in to Office 365.
Convert-MsolDomainToFederated	The Convert-MsolDomainToFederated cmdlet converts the specified domain from standard authentication to single sign-on (also known as identity federation), including configuring the relying party trust settings between the Active Directory Federation Services 2.0 server and Office 365. As part of converting a domain from standard authentication to single sign-on, each user must also be converted. This conversion happens automatically the next time a user signs in; no action is required by the administrator.
Get-MsolFederationProperty	The Get-MsolFederationProperty cmdlet gets key settings from both the Active Directory Federation Services 2.0 server and Office 365. You can use this information to troubleshoot authentication problems caused by mismatched settings between the Active Directory Federation Services 2.0 server and Office 365.
Get-MsolDomainFederationSettings	The Get-MsolDomainFederationSettings cmdlet gets key settings from Office 365. Use the Get-MsolFederationProperty cmdlet to get settings for both Office 365 and the Active Directory Federation Services server.
Remove-MsolFederatedDomain	The Remove-MsolFederatedDomain cmdlet removes the specified single sign-on domain from Office 365 and the associated relying party trust settings in Active Directory Federation Services 2.0. Note: If the domain specified has objects associated with it, you will not be able to remove the domain.
Set-MsolDomainFederationSettings	The Set-MsolDomainFederationSettings cmdlet is used to update the settings of a single sign-on domain.
Set-MsolADFSContext	The Set-MsolADFSContext cmdlet sets the credentials to connect to Office 365 and to the Active Directory Federation Services 2.0 (AD FS 2.0) server. This cmdlet must be run before making other single sign-on (also known as identity federation) cmdlet calls. If this cmdlet is called without parameters, the user will be prompted for credentials to connect to the different systems. When the AD FS 2.0 server is used remotely, the user must specify the computer name of the primary AD FS 2.0 server. Note that the specified logfile is shared by all single sign-on cmdlets for the session. A default logfile is created if one is not specified.
Update-MsolFederatedDomain	The Update-MsolFederatedDomain cmdlet changes settings in both the Active Directory Federation Services 2.0 server and Office 365. It is necessary to run this cmdlet whenever the URLs or certificate information within Active Directory Federation Services 2.0 change due to configuration changes or through regular maintenance of the certificates, such as when a certificate is about to expire. This cmdlet should also be run when changes occur in Office 365. To confirm that the information in the two systems is correct, the Get-MsolFederationProperty cmdlet can be used to retrieve the settings.

Step-6: Subscription/License

Manage subscriptions and licenses

Use the following cmdlets to manage subscriptions, accounts, and licenses.

Use the following cmdlets to perform tasks related to managing your company's information and connecting to Microsoft Office 365 for enterprises. There are also cmdlets for tasks performed by partner companies.

Windows PowerShell cmdlet	Description
Get-MsolSubscription	The Get-MsolSubscription cmdlet returns all the subscriptions that the company has purchased. When assigning licenses to users, the Get-MsolAccountSku API should be used instead.
Get-MsolAccountSku	The Get-MsolAccountSku will return all the SKUs that the company owns.
New-MsolLicenseOptions	The New-MsolLicenseOptions cmdlet creates a new License Options object. This cmdlet disables specific service plans when assigning a user a license using the Add-MsolUser and Set-MsolUserLicense cmdlets.
Set-MsolUserLicense	The Set-MsolUserLicense cmdlet can be used to adjust the licenses for a user. This can include adding a new license, removing a license, updating the license options, or any combination of these actions.
Connect-MsolService	The Connect-MsolService cmdlet will attempt to initiate a connection to Office 365. The caller must either provide their credential (a PSCredential object), or use the UseCurrentCredential option if the current logged in user is federated with Office 365. This cmdlet may return a warning or error if the version of the module being used is out of date.
Set-MsolDirSyncEnabled	The Set-MsolDirSyncEnabled cmdlet is used to turn directory synchronization on or off for a company.
Get-MsolPartnerContract	The Get-MsolPartnerContract cmdlet should only be used by partners, as it is used to retrieve a list of contracts for a partner. The input to this cmdlet should be a domain to look up, which must be verified for the tenant. If the company exists and the partner has access to this company, then the corresponding contract will be returned.
Get-MsolPartnerInformation	The Get-MsolPartnerInformation cmdlet is used to retrieve partner-specific information. This cmdlet should only be used for partner tenants.
Set-MsolPartnerInformation	The Set-MsolPartnerInformation cmdlet is used by partners to set partner-specific properties. These properties will be viewable by all tenants that the partner has access to.
Get-MsolContact	The Get-MsolContact cmdlet can be used to retrieve a contact object, or list of contacts. A single contact will be retrieved if the ObjectId parameter is used.
Remove-MsolContact	The Remove-MsolContact cmdlet is used to delete a contact from the Microsoft Online directory.
Get-MsolCompanyInformation	The Get-MsolCompanyInformation cmdlet will retrieve company-level information.
Set-MsolCompanyContactInformation	The Set-MsolCompanyContactInformation cmdlet is used to set company-level contact preferences. This includes email addresses for billing, marketing, and technical notifications about Office 365.
Set-MsolCompanySettings	The Set-MsolCompanySettings cmdlet is used to set company-level configuration settings.
Redo-MsolProvisionContact	The Redo-MsolProvisionContact cmdlet can be used to retry the provisioning of a contact object in the Microsoft Online directory when a previous attempt to create the contact object resulted in an error.
Redo-MsolProvisionGroup	The Redo-MsolProvisionGroup cmdlet can be used to retry the provisioning of a group object in the Microsoft Online directory when a previous attempt to create the group object resulted in an error.
Redo-MsolProvisionUser	The Redo-MsolProvisionUser cmdlet can be used to retry the provisioning of a user object in the Microsoft Online directory when a previous attempt to create the user object resulted in an error.

Step-7: Exchange Online

Use Exchange Online cmdlets to perform management tasks that aren't available or practical in the Exchange Control Panel. For example, you can create dynamic distribution groups, create or update many user accounts at one time, and script automated solutions. For a list of the Exchange Online cmdlets that are currently available to administrators, see Reference to Available Windows PowerShell Cmdlets.

More on: Use Windows PowerShell to manage Office 365

Microsoft Certificates:

- Introduction to Selling Microsoft Online Services to Partners
- MPN Technical assessment for Microsoft Office 365
- MPN Sales and Service Assessment for Microsoft Online Services
- MOSPA – Microsoft Online Service Partner Agreement

Cloud Power Windows Azure Microsoft Partner Network

Thank You

Muthu Swamy S, APSM, PMP, CSM, PM-2008
rms.reddy@hotmail.com
http://linkedin.com/in/msbgl