## Limitations and pre-requirements

1. Install following PowerShell Modules on Server:

* Az
* Microsoft.Online.SharePoint.PowerShell
* MSOnline
* SharePointPnPPowerShellOnline
* Microsoft.Graph.Intune

To install a module, run PowerShell as administrator, and perform a command:

Install-Module *Module-Name*

1. For each tenant following information (currently two tenants are in Tenants table) required:

For All tasks except Azure Resources (Resource Group, VNet, VM etc)

* <guid> TenantID. '698e87df-9f6c-451b-acfd-09fd709efd86'
* <string> DomainName. 'logiwayeu'
* <string> AdminUser. 'cloudadmin@logiwayeu.onmicrosoft.com'
* <string> AdminPassword. 'Vab0!240EUA'
* <string> AppId (Security Principal). '6a86542a-2144-43ce-a5e1-50f135ccc7aa'
* <string> AppPassword. 'Passw0rd!'

For Azure Resources (Resource Group, VNet, VM etc)

* <guid> TenantID. '3b811c12-df4a-41b3-834a-5b9420089c1b'
* <string> DomainName. 'thomasschmitzlogiway'
* <string> AdminUser. 'admin@thomasschmitzlogiway.onmicrosoft.com'
* <string> AdminPassword. 'Wasser9!'

1. To create Service Principal App, use 01.Create.Service.Principal.ps1 script from preworks folder.
2. Personal site URLs (OneDrive) for User under specific Tenant are cached in SQL table to avoid excessive API calls

## Database

Database is an Azure SQL DB, with following properties:

* Server: psqueue.database.windows.net
* User: dbadmin
* Password: s52fG6!Av
* Connection string: Server=tcp:psqueue.database.windows.net,1433;Initial Catalog=PSQueue;Persist Security Info=False;User ID=dbadmin;Password=s52fG6!Av;MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;

It contains following tables:

* Queue (should be filled by php script). Values in bold are required and filled by requestor.
  + ID – int, autoincrement
  + **RequestID** – string, must be unique
  + **TenantID** – string, guid with tenant id.
  + **Command** – string, one of command listed below, e.g. Add-LwPnpFolder or New-LwAzVM
  + **Params** – string, json-formatted parameters of the command
  + **AllowParallel** – smallint, whether command can be processed in parallel with others (1) or not (0). Nornally Get-XXX commands can be processed such way.
  + Response – string, return value
  + RetryCount – int, used by retry mechanisms
  + Status – int. 0 – not processed, 1 – successfully processed, 2 – processed with error, 3 – currently processing
  + CreatedTime – datetime
  + ProcessedTime – datetime
  + PackageID – string, used internally
* Tenants (should be filled by admin)
* PersonalSites (used for caching)

Use PSQueue.sql to create tables.

Use PSQueueData.sql to fill with initial (logiway) tenant.

## Files

### AppSettings.json

Configuration for Robot. Contains connection string and timeout between pool requests.

### Robot.ps1

The script pulls tasks from the table Queue, processes them in asynchronous mode.

Tasks can be processed in parallel if they are:

* Marked with “AllowParallel” flag
* Are from different tenants.

Just run .\Robot.ps1 in PowerShell window and leave the process work.

### Enqueue.ps1

To enqueue command to robot.

Parameters:

* [string] Command
* [hashtable] Params

Sample:

.\ Enqueue.ps1 -Command 'Add-LwPnpFolder' -AsJson - TenantId '698e87df-9f6c-451b-acfd-09fd709efd86' -Params @{ SiteUrl = $SiteUrl; Folder = 'Freigegebene Dokumente'; Name = 'Test-Folder'}

### Invoke-LwCommand.ps1

To perform command locally.

Parameters:

* [string] TenantId
* [string] Command
* [hashtable] Params
* [switch] AsJson (optional)
* [switch] Compress (optional)

Sample:

.\Invoke-LwCommand.ps1 -Command 'Add-LwPnpFolder' -AsJson -Params @{ TenantId = $tenantId; SiteUrl = $SiteUrl; Folder = 'Freigegebene Dokumente'; Name = 'Test-Folder'}

### Test-LwAzure.ps1

A lot of samples to perform commands locally. To enqueue to DB, just replace Invoke-LwCommand.ps1 with Enqueue.ps1.

## Commands

Mandatory parameters are in bold

### New-LwSPOSite

Parameters: <string> **Url**, <string> Title, <string> **Owner**, <int64> StorageQuota, <int> CompatibilityLevel, <int> LocaleId, <bool> NoWait , <double> ResourceQuota, <string> Template, <int> TimeZoneId

Owner – Site Collection primary owner (e-mail address)

StorageQuota – Storage quota in megabytes

CompatibilityLevel – Version of templates to use

ResourceQuota – Quota in Sandboxed Solutions units

Template – Template type. Use the Get-SPOWebTemplate

Example: {"Url":"https://logiwayeu.sharepoint.com/sites/AMTest", "Owner":"cloudadmin@logiwayeu.onmicrosoft.com", "Template":"STS#3", "StorageQuota":1000}

### Get-LwSPOSite

Parameters: <string> **Url**, <bool> DisableSharingForNonOwnersStatus, <bool> Detailed, <string> Limit, <string> Filter, <bool> IncludePersonalSite, <string> Template

DisableSharingForNonOwnersStatus – Prevents non-owner from sharing

Detailed – Get additional property information on a site collection

Limit – Maximum number of site collections to return. It can be any number. To retrieve all site collections, use ALL. The default value is 200

Filter – Script block of the server-side filter to apply

Template – Displays sites of a specific template. For example, STS, STS#0 or STS#1

Example: {"Url":"https://logiwayeu.sharepoint.com/sites/AMTest"}

### Remove-LwSPOSite

Parameters: <string> **Url**, <bool> NoWait, <bool> Permanently

NoWait – Continue executing script immediately

Permanently – Remove also from recycle bin

Example: {"Url":"https://logiwayeu.sharepoint.com/sites/AMTest", "Permanently":true}

### Get-LwSPOPersonalSiteForUser

Parameters: <string> **UPN**

Example: {"UPN":" test@logiway.eu"}

### Add-LwPnpFile

Note: this command can not be used in web enqueue scenario, because file located on user computer is not accessible nor through path either as stream.

Parameters: <string> **SiteUrl** OR <string> **User**, <string> **Path** OR <stream> Stream, <string> **Folder**, <string> NewFileName, <bool> Checkout; <string> CheckInComment, <bool> Approve, <string> ApproveComment, <bool> Publish, <string> PublishComment, <hashtable> FieldValues

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Path/Stream – Local file path/File stream. Only one of them should be specified

Folder – Site folder including list identity

NewFileName – File name after upload

Checkout – If versioning is enabled, this will check out the file first if it exists, upload the file, then check it in again

Approve – Will auto approve the uploaded file

Publish – Will auto publish the file

Example: {"User":"cloudadmin@logiwayeu.onmicrosoft.com", "Path":".\\Test-LwPnPOneDrive.ps1", "Folder":"Documents", "NewFileName":"Test-LwPnPNew1.ps1", "Checkout":true, "CheckInComment":"CheckIn: Uploaded by PowerShell"}

### Add-LwPnpFolder

Parameters: <string> **SiteUrl** OR <string> **User**, <string> **Folder**, <string> **Name**

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder – The parent folder in the site including list identity

Name –The folder name

Example: {"SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente", "Name":"Test-Folder"}

### Break-LwPnPRoleInheritance

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File**, <bool> CopyRoleAssignments, <bool> ClearSubscopes

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

Example: { "SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente/Test-Folder}

### Reset-LwPnPRoleInheritance

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File**

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

Example: { "SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente/Test-Folder}

### Add-LwPnPRoleAssignment

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File**, <bool> CopyRoleAssignments, <bool> ClearSubscopes, <string> **Identity**, <string> **RoleName**

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

Identity – User or group UPN

Example: {"SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente/Test-Folder", "Identity":"test@logiway.eu", "RoleName":"Contribute"}

### Remove-LwPnPRoleAssignment

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File**, <string> **Identity**

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

Identity – User or group UPN

Example: {"SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente/Test-Folder", "Identity":"test@logiway.eu"}

### Get-LwPnPRoleAssignments

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File**

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

Example: {"SiteUrl":"https://logiwayeu.sharepoint.com/sites/AMTest", "Folder":"Freigegebene Dokumente/Test-Folder"}

### Add-LwPnpShare

Parameters: <string> **SiteUrl** OR <string> **User,** <string> **Folder** OR <string> **File,** <string> **UPN**, <bool> IsGuestUser, <string> Role, <bool> AnonymousLink, <bool> PropageAcl, <bool> SendEmail, <string> EmailSubject

SiteUrl is used for SharePoint User for OneDrive. Only one of them should be specified

Folder/File – Folder/File name including list identity. Only one of them should be specified

UPN – The principal name (UPN) of the user

IsGuestUser – Is guest user (true) or unvalidated email (false)

Role – 'Edit' or 'View' (default) permission

AnonymousLink – If an e-mail is being sent, this determines if an anonymous link should be added to the message

PropageAcl – A flag to determine if permissions should be pushed to items with unique permissions

SendEmail – Flag to determine if an e-mail notification should to sent, if e-mail is configured

EmailSubject – Mail subject. Required only if not anonymous link'

Example: {"User":"cloudadmin@logiwayeu.onmicrosoft.com", "File":"Documents/share-guest.txt", "UPN":"gen444ral@mail.ru", "IsGuestUser":true, "Role":"Edit", "SendEmail":true, "EmailSubject":"There is a document for you"}

### Get-LwAzureUser

Parameters: <guid> ObjectId, <string> UPN

Specify whether ObjectId or UPN, or nothing – all users will be retrieved then

Example: {"UPN":"john\_smith@logiway.eu"}

### New-LwAzureUser

Parameters: <string> UPN, <string> Surname, <string> GivenName, <string> DisplayName, <string> MailNickName, <string> **Password**, <nullable<bool>> ForceChangePasswordNextLogin, <nullable<bool>> EnforceChangePasswordPolicy, <bool> DisableStrongPassword, <bool> PasswordNeverExpires, <string> UserType, <bool> Enabled, <bool> LocalAccount, <string> ImmutableId, <string> Country, <string> State, <string> PostalCode, <string> City, <string> StreetAddress, <string> TelephoneNumber, <string> Mobile, <string> FacsimileTelephoneNumber, <hashtable> ExtProps

DisplayName – if not specified, will be combined from GivenName and Surname

UPN – is required for a work or school account

MailNickName – Mail nick name, if not specified, will be retrieved from UPN

EnforceChangePasswordPolicy – Enforce to change password policy on user creation

DisableStrongPassword – Allow weaker passwords than the default policy to be specified. Default is false.

UserType – A string value that can be used to classify user types in your directory, such as "Member" (default) and "Guest".

Enabled – Whether account is enabled. Default is true.

LocalAccount – Whether is a local account for an Azure Active Directory B2C tenant (default)

ImmutableId – Is used to associate an on-premises Active Directory user account to their Azure AD user object. This property must be specified when creating a new user account in the Graph if you are using a federated domain for the user's userPrincipalName (UPN) property.

Example: {"UPN":"john\_smith@logiway.eu", "Password":"Passw0rd!", "DisplayName":"John Smith"}

### Update-LwAzureUser

Parameters: <guid> ObjectId, <string> UPN, <string> Surname, <string> GivenName, <string> DisplayName, <string> MailNickName, <string> Password, <nullable<bool>> ForceChangePasswordNextLogin, <nullable<bool>> EnforceChangePasswordPolicy, <nullable<bool>> DisableStrongPassword, <nullable<bool>> PasswordNeverExpires, <string> UserType, <nullable<bool>> Enabled, <nullable<bool>> LocalAccount, <string> ImmutableId, <string> Country, <string> State, <string> PostalCode, <string> City, <string> StreetAddress, <string> TelephoneNumber, <string> Mobile, <string> FacsimileTelephoneNumber, <hashtable> ExtProps

### Remove-LwAzureUser

Parameters: <guid> ObjectId, <string> UPN

Specify whether ObjectId or UPN

Example: {"UPN":"john\_smith@logiway.eu"}

### Set-LwAzUserPassword

Parameters: <string> **UPN**, <string> **Password**, <bool> ForceChangePasswordNextLogin

Example: {"UPN":"john\_smith@logiway.eu", "Password":"Ph5ok!hj"}

Enable-LwAzUser

Parameters: <string> **UPN**, <bool> Enabled

Enabled – whether to enable (default) or disable the user

Example: {"UPN":"john\_smith@logiway.eu", "Enabled":true}

### Get-LwAzureGroup

Parameters: <guid> ObjectId, <string> UPN

Specify ObjectId or nothing – all groups will be retrieved then

Example: {"ObjectId":" 5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### New-LwAzureGroup

Parameters: <string> **DisplayName**, <string> Description, <string> GivenName, <bool> MailEnabled, <string> **MailNickName**, <bool> SecurityEnabled

Example: {"DisplayName":"TestGroup", "Description": "PowerShell Tests", "MailEnabled": false, "MailNickName": "sample", "SecurityEnabled": true}

### Update-LwAzureGroup

Parameters: <guid> ObjectId, <string> DisplayName, <string> Description, <string> GivenName, <bool> MailEnabled, <string> MailNickName, <bool> SecurityEnabled

Example: {"ObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0", "DisplayName":"TestGroup2"}

### Remove-LwAzureGroup

Parameters: <guid> **ObjectId**

Example: {"ObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### Get-LwAzureGroupMember

Parameters: <guid> **ObjectId**

Example: {"ObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### Add-LwAzureGroupMember

Parameters: <guid> **ObjectId**, <guid> **RefObjectId**

Example: {"ObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0", "RefObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### Remove-LwAzureGroupMember

Parameters: <guid> **ObjectId**, <guid> **MemberId**

Example: {"ObjectId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0", "MemberId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### Get-LwAzureDevice

Parameters: <guid> ObjectId, <string> DisplayName

Specify ObjectId or DisplayName, or nothing – all devices will be retrieved then

Example: {"ObjectId":"a9582f31-12b4-402d-b5f6-f64e504e986d"}

### New-LwAzureDevice

Parameters: <string> **DisplayName**, <guid> **DeviceId**, <string> **DeviceOSType**, <string> DeviceOSVersion, <int> DeviceObjectVersion, <bool> AccountEnabled, <bool> IsCompliant, <bool> IsManaged, <hashtable[]> AlternativeSecurityIds, <datetime> ApproximateLastLogonTimeStamp, <string[]> DevicePhysicalIds, <string> DeviceMetadata

DisplayName - display name of the new device

DeviceId - ID of the device

DeviceOSType - operating system type of the new device

DeviceOSVersion - operating system version of the new device

DeviceObjectVersion - object version of the device

AccountEnabled - whether the account is enabled

IsCompliant - true if the device complies with Mobile Device Management (MDM) policies; otherwise, false

IsManaged - true if the device is managed by a Mobile Device Management (MDM) app such as Intune; otherwise, false

AlternativeSecurityIds – array of hashtables with alternate security ids. Properties: <string> IdentityProvider, <string> Type, <byte[]> Key.

DeviceMetadata - Metadata for this device

Example: {"DisplayName": "ANDREI-W540", "DeviceId":"d224105d-5f46-4fd4-954a-2bb0c6e303d8", "DeviceOSType":"Windows", "DeviceOSVersion":"10.0.17134.0", "DeviceObjectVersion":2, "AccountEnabled":true, "AlternativeSecurityIds":[{"Type":2, "Key":[88,0,53,0,48,0,57,0,58,0,60,0,83,0,72,0,65,0,49,0,45,0,84,0,80,0,45,0,80,0,85,0,66,0,75,0,69,0,89,0,62,0,69,0,49,0,48,0,67,0,51,0,52,0,54,0,68,0,66,0,52,0,68,0,50,0,52,0,53,0,54,0,52,0,51,0,48,0,65,0,55,0,67,0,69,0,51,0,67,0,55,0,57,0,57,0,68,0,65,0,48,0,68,0,51,0,49,0,68,0,53,0,65,0,68,0,54,0,53,0,65,0,73,0,116,0,116,0,106,0,117,0,76,0,117,0,116,0,88,0,103,0,87,0,78,0,113,0,81,0,72,0,108,0,78,0,106,0,106,0,88,0,69,0,84,0,86,0,76,0,50,0,84,0,115,0,120,0,120,0,48,0,114,0,114,0,105,0,69,0,75,0,89,0,115,0,88,0,68,0,99,0,43,0,90,0,52,0,61,0] }], "DevicePhysicalIds":["[USER-GID]:a4b7ab35-8a83-4102-b773-6c4f63b54448:6825773979753362","[GID]:g:6825773979753362","[USER-HWID]:a4b7ab35-8a83-4102-b773-6c4f63b54448:6755410885010397","[HWID]:h:6755410885010397"]}

### Remove-LwAzureDevice

Parameters: <guid> **ObjectId**

Example: {"ObjectId":"a9582f31-12b4-402d-b5f6-f64e504e986d"}

### Get-LwOfficeGroup

Parameters: <string> **Identity**

Identity – Group identity (name, alias, DN, email, guid)

Example: {"Identity":" MyMail"}

### Get-LwIntuneDeviceConfigurationPolicy

Parameters: <guid> PolicyId, <string> PolicyName

Specify PolicyId or PolicyName, or nothing – all policies will be retrieved then

Example: {"PolicyName":"AMFavorities"}

### New-LwIntuneDeviceConfigurationPolicy

Parameters: <string> **DisplayName**, <string> Description, <string> **ConfigurationType**, <hashtable> **Settings**

ConfigurationType - predefined type, for example windows10GeneralConfiguration or androidGeneralDeviceConfiguration.

Settings – hashtable with settings for policy

Example: {"DisplayName":"BlockCopy", "ConfigurationType":"windows10GeneralConfiguration", "Settings":{"copyPasteBlocked":true}}

### Update-LwIntuneDeviceConfigurationPolicy

Parameters: <guid> **PolicyId**, <string> DisplayName, <string> Description, <string> **ConfigurationType**, <hashtable> Settings

ConfigurationType - predefined type, for example windows10GeneralConfiguration or androidGeneralDeviceConfiguration.

Settings – hashtable with settings for policy

Example: {"PolicyId":"1a322d9d-898f-4631-a30c-a901a8e9cf62", "DisplayName":"BlockCopyNew", "ConfigurationType":"windows10GeneralConfiguration", "Settings":{"copyPasteBlocked":false}}

### Remove-LwIntuneDeviceConfigurationPolicy

Parameters: <guid> **PolicyId**

Example: {"PolicyId":"1a322d9d-898f-4631-a30c-a901a8e9cf62"}

### Get-LwIntuneManagedDevice

Parameters: <guid> DeviceId

Specify DeviceId, or nothing – all devices will be retrieved then

Example: {" DeviceId":"110b4e49-bd79-48c0-94ec-3b433d41fbcc "}

### Get-LwIntuneDeviceCompliancePolicyAssignment

Parameters: <guid> **PolicyId**

Example: {"PolicyId":"35b54cd1-f6a4-4efa-936a-e1b9d17f432b"}

### New-LwIntuneDeviceCompliancePolicyAssignment

Parameters: <guid> **PolicyId**, <guid> **GroupId**

Example: {"PolicyId":"35b54cd1-f6a4-4efa-936a-e1b9d17f432b", "GroupId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

### Remove-LwIntuneDeviceCompliancePolicyAssignment

Parameters: <guid> **PolicyId**, <guid> **GroupId**

Example: {"PolicyId":"35b54cd1-f6a4-4efa-936a-e1b9d17f432b", "GroupId":"5e7134d8-4edd-4cfe-a21f-585b0f4ed6d0"}

## Invoke-LwIntuneManagedDeviceSyncDevice

Forces device sync in Intune.

Parameters: <guid> **DeviceId**

Example: {"DeviceId":"110b4e49-bd79-48c0-94ec-3b433d41fbcc"}

### New-LwOfficeGroup

Parameters: <string> **DisplayName**,<string> **Alias**,<string> AccessType, <string[]> Members, <string> Notes, <string> Owner, <bool> AutoSubscribeNewMembers

DisplayName – group display name

Alias – mail nick name

AccessType – privacy type for the Office 365 Group <Public | Private>

Members – Group members' identities (name, alias, DN, email, guid)

Notes – Description of the Office 365 Group

Owner – An owner is a group member who has certain privileges, such as the ability to edit group properties

AutoSubscribeNewMembers – whether to automatically subscribe new members that are added to the Office 365 Group to conversations and calendar events.

Example: {"Alias":"MyMail", "DisplayName":"OfGroup1"}

### Update-LwOfficeGroup

Parameters: <string> **DisplayName**,<string> **Alias**,<string> AccessType, <string[]> Members, <string> Notes, <string> Owner, <bool> AutoSubscribeNewMembers, <string[]> AcceptMessagesOnlyFromSendersOrMembers, <bool> AlwaysSubscribeMembersToCalendarEvents, <bool> CalendarMemberReadOnly, <bool> SubscriptionEnabled, <string[]> RejectMessagesFromSendersOrMembers, <bool> UnifiedGroupWelcomeMessageEnabled

DisplayName – group display name

Alias – mail nick name

AccessType – privacy type for the Office 365 Group <Public | Private>

Members – Group members' identities (name, alias, DN, email, guid)

Notes – Description of the Office 365 Group

Owner – An owner is a group member who has certain privileges, such as the ability to edit group properties

AutoSubscribeNewMembers – whether to automatically subscribe new members that are added to the Office 365 Group to conversations and calendar events.

AcceptMessagesOnlyFromSendersOrMembers - Who is allowed to send messages to this recipient. Messages from other senders are rejected.

AlwaysSubscribeMembersToCalendarEvents -- Default subscription settings of new members that are added to the Office 365 Group

CalendarMemberReadOnly - Whether to set read-only Calendar permissions to the Office 365 Group for members of the group.

SubscriptionEnabled - Specifies whether subscriptions to conversations and calendar events are enabled for the Office 365 Group.

RejectMessagesFromSendersOrMembers - Who isn't allowed to send messages to this recipient. Messages from these senders are rejected.

UnifiedGroupWelcomeMessageEnabled - Whether to enable or disable sending system-generated welcome messages to users who are added as members to the Office 365 Group.

Example: {"Alias":"MyMail", "DisplayName":"OfGroup1ABC", "AutoSubscribeNewMembers":false, "SubscriptionEnabled":false}

### Remove-LwOfficeGroup

Parameters: <string> **Identity**

Identity – Group identity (name, alias, DN, email, guid)

Example: {"Identity":"MyMail"}

### Get-LwOfficeGroupMembers

Parameters: <string> **Identity**, <string> Type

Identity – Group identity (name, alias, DN, email, guid)

Type – Members' type (Members | Owners | Subscribers | Aggregators)

Example: {"Identity":"MyMail", "Type":"Members"}

### Add-LwOfficeGroupMembers

Parameters: <string> **Identity**, <string[]> **Members**, <string> Type

Identity – Group identity (name, alias, DN, email, guid)

Members – Group members' identities (name, alias, DN, email, guid)

Type – Members' type (Members | Owners | Subscribers | Aggregators)

Example: {"Identity":"MyMail", "Members":["test@logiway.eu"], "Type":"Members"}

### Remove-LwOfficeGroupMembers

Parameters: <string> **Identity**, <string[]> **Members**, <string> Type

Identity – Group identity (name, alias, DN, email, guid)

Members – Group members' identities (name, alias, DN, email, guid)

Type – Members' type (Members | Owners | Subscribers | Aggregators)

Example: {"Identity":"MyMail", "Members":["test@logiway.eu"], "Type":"Members"}

### Get-LwMsolUsersReport

Parameters: <none>

Get-LwAzResourceGroup

Parameters: <string> Name

Example: {"Name":"AMLogiway"}

New-LwAzResourceGroup

Parameters: <string> **Name**, <string> **Location**

Location – Region in which to create

Example: {"Name":"AMLogiway", "Location":"West Europe"}

### Get-LwAzStorageAccount

Parameters: <string> Name, <string> ResourceGroupName

Example: {"Name":"amlogiway", "ResourceGroupName":"AMLogiway"}

### New-LwAzStorageAccount

Parameters: <string> **Name**, <string> Location, <string> **ResourceGroupName**, <string> SkuName, <string> Kind, <NetworkRuleSet>

Location – Region in which to create

ResourceGroupName – Name of the resource group

SkuName – one of 'Standard\_LRS', 'Standard\_ZRS', 'Standard\_GRS', 'Standard\_RAGRS', 'Premium\_LRS'

Kind – one of 'Storage', 'StorageV2', 'BlobStorage'

Example: {"Name":"amlogiway", "Location":"West Europe", "ResourceGroupName":"AMLogiway", "SkuName":"Standard\_LRS"}

### Get-LwAzNetworkSecurityGroup

Parameters: <string> Name, <string> ResourceGroupName

Example: {"Name":"AMLogiway"}

### New-LwAzNetworkSecurityGroup

Parameters: <string> Name, <string> Location, <string> **ResourceGroupName**, <hashtable[]> SecurityRules, [hashtable] Tag

Location – Region in which to create

ResourceGroupName – Name of the resource group

SecurityRules – array of settings in form { <string> **Name**, <string> Description, <string> Protocol, <string[]> SourcePortRange, <string[]> DestinationPortRange, <string[]> SourceAddressPrefix, <string[]> DestinationAddressPrefix, <string> Access, <int> Priority, <string] Direction }

Tag – Key-value pairs in the form of a hash table. For example: {key0="value0", key1=null, key2="value2"}

Example: {"Name":"AMLogiway", "Location": "West Europe", "ResourceGroupName": "AMLogiway"; "SecurityRules":[{"Name":"rdp-rule", "Description":"Allow RDP", "Protocol":"Tcp", "SourcePortRange":"\*", "DestinationPortRange":"3389", "SourceAddressPrefix":"\*", "DestinationAddressPrefix":"\*", "Access":"Allow", "Priority":100, "Direction":"Inbound"}, {"Name":"www-rule", "Description":"Allow WWW", "Protocol":"Tcp", "SourcePortRange":"\*", "DestinationPortRange":"80", "SourceAddressPrefix":"\*", "DestinationAddressPrefix":"\*", "Access":"Allow", "Priority":101, "Direction":"Inbound"}]}

### Get-LwAzVirtualNetwork

Parameters: <string> Name, <string> ResourceGroupName

Name – Name of the virtual network

ResourceGroupName – Name of the resource group

Example: {"Name":"AMLogiway", "ResourceGroupName":"AMLogiway"}

### New-LwAzVirtualNetwork

Parameters: <string> **Name**, <string> **Location**, <string> **ResourceGroupName**, <string[]> **AddressPrefix**, <string[]> DnsServer, <hashtable[]> Subnets, <hashtable> Tag, <bool> EnableDdosProtection, <string> DdosProtectionPlanId

Name – Name of the virtual network

ResourceGroupName – Name of the resource group

Subnets – list of subnets in form of hashes {<string> Name, <string[]> AddressPrefix, <string> NetworkSecurityGroup, <string[]> ServiceEndpoint)

Example: {"Name":"AMLogiway", "ResourceGroupName":"AMLogiway", "Prefix":"10.0.0.0/16", "Subnets":[{"Name":"FrontendSubnet","AddressPrefix":"10.0.1.0/24", "NetworkSecurityGroup":"AMLogiway-NSG"}, {"Name":"BackendSubnet", "AddressPrefix":"10.0.2.0/24", "NetworkSecurityGroup":"AMLogiway-NSG"}]}

### Get-LwAzPublicIpAddress

Parameters: <string> Name, <string> ResourceGroupName

Example: {"Name":"AMLogiwayIP", "ResourceGroupName":"AMLogiway"}

### New-LwAzPublicIpAddress

Parameters: <string> **Name**, <string> **Location**, <string> **ResourceGroupName**, <string> Sku, <string> AllocationMethod, <string> IpAddressVersion, <string> DomainNameLabel, <string> ReverseFqdn, <int> IdleTimeoutInMinutes, <string[]> Zone, <hashtable> Tag

Name - Name of the public IP address

Location - Region in which to create a public IP address

ResourceGroupName - Name of the resource group in which to create a public IP address

Sku - The public IP Sku name

AllocationMethod - Method with which to allocate the public IP address. The acceptable values for this parameter are: Static or Dynamic (default)

IpAddressVersion - Version of the IP address, IPv4 (default) or IPv6

DomainNameLabel - Relative DNS name for a public IP address

ReverseFqdn - Reverse fully qualified domain name (FQDN)

IdleTimeoutInMinutes - Idle time-out, in minutes

Zone - A list of availability zones denoting the IP allocated for the resource needs to come from

Tag - Key-value pairs in the form of a hash table. For example: @{key0="value0";key1=$null;key2="value2"}

Example: {"Name":"AMLogiwayIP", "Location": "West Europe", "ResourceGroupName":"AMLogiway", "AllocationMethod":"Static", "IdleTimeoutInMinutes":4}

### New-LwAzVM

Parameters: <string> **Name**, <string> **Location**, <string> **ResourceGroupName**, <string> Size, <string> AvailabilitySetId, <string> LicenseType, <hashtable> Tags, <string[]> Zone, <bool> EnableUltraSSD, <string> OS, <string> ComputerName, <string> **User**, <string> **Password**,<bool> WinRMHttp,<bool> WinRMHttps, <string> PublisherName, <string> Offer, <string> Skus, <string> Version, <hashtable> OSDisk, <bool> BootDiagnostics, <string> StorageAccountName, <hashtable[]> NetworkInterfaces

Name - Name for the virtual machine

Location - region for a network interface

ResourceGroupName - name of the resource group in which to create a public IP address

Size - size for the virtual machine

AvailabilitySetId - ID of an availability set. To obtain an availability set object, use the Get-AzAvailabilitySet cmdlet. The availability set object contains an ID property

LicenseType - the license type, which is for bringing your own license scenario'

Tags - the tags attached to the resource

Zone - the availability zone list for the virtual machine. The allowed values depend on the capabilities of the region. Allowed values will normally be 1,2,3'

EnableUltraSSD - Enables a capability to have one or more managed data disks with UltraSSD\_LRS storage account type on the VM. Managed disks with storage account type UltraSSD\_LRS can be added to a virtual machine only if this property is enabled

Operating system - Operating system. Values are Windows (default) and Linux.

ComputerName – computer name

User - Admin user name

Password - Admin user password

WinRMHttp - This operating system uses HTTP WinRM

WinRMHttps - This operating system uses HTTPS WinRM

PublisherName - The name of a publisher of a VMImage. To obtain a publisher, use the Get-AzVMImagePublisher cmdlet. Default is 'MicrosoftWindowsServer'.

Offer - The type of VMImage offer. To obtain an image offer, use the Get-AzVMImageOffer cmdlet. Default is 'WindowsServer'.

Skus - VMImage SKU. To obtain SKUs, use the Get-AzVMImageSku cmdlet.

Version - Version of a VMImage. To use the latest version, specify a value of latest instead of a particular version. Default is 'latest'.

OSDisk – **hashtable with OS Disk data**

$cmd = 'Set-AzVMOSDisk -VM $vm'

if ($OSDisk.Name) { $cmd += ' -Name $OSDisk.Name'}

if ($OSDisk.VhdUri) { $cmd += ' -VhdUri $OSDisk.VhdUri'}

if ($OSDisk.Caching) { $cmd += ' -Caching $OSDisk.Caching'}

if ($OSDisk.SourceImageUri) { $cmd += ' -SourceImageUri $OSDisk.SourceImageUri'}

if ($OSDisk.CreateOption) { $cmd += ' -CreateOption $OSDisk.CreateOption'}

if ($OS -eq 'Windows') { $cmd += ' -Windows'}

if ($OS -eq 'Linux') { $cmd += ' -Linux'}

if ($OSDisk.DiskSizeInGB) { $cmd += ' -DiskSizeInGB $OSDisk.DiskSizeInGB'}

if ($OSDisk.ManagedDiskId) { $cmd += ' -ManagedDiskId $OSDisk.ManagedDiskId'}

if ($OSDisk.StorageAccountType) { $cmd += ' -StorageAccountType $OSDisk.StorageAccountType'}

if ($OSDisk.WriteAccelerator) { $cmd += ' -WriteAccelerator'}

if ($OSDisk.DiffDiskSetting) { $cmd += ' -DiffDiskSetting $OSDisk.DiffDiskSetting'}

BootDiagnostics - Enable or disable boot diagnostics.

StorageAccountName - Name of the storage account in which to save boot diagnostics data

NetworkInterfaces - **Array with Network interfaces**

[Parameter(Mandatory=$true, HelpMessage='Name of the network interface to create')] [string] $Name,

[Parameter(Mandatory=$true, HelpMessage='Region for a network interface')] [string] $Location,

[Parameter(Mandatory=$true, HelpMessage='Name of the resource group in which to create a public IP address')] [string] $ResourceGroupName,

[Parameter(Mandatory=$true, HelpMessage='ID of the subnet for which to create a network interface')] [string] $SubnetId,

[Parameter(Mandatory=$false, HelpMessage='ID of a PublicIPAddress object to assign to a network interface')] [string] $PublicIpAddressId,

[Parameter(Mandatory=$false, HelpMessage='ID of a network security group')] [string] $NetworkSecurityGroupId,

[Parameter(Mandatory=$false, HelpMessage='Static IPv4 IP address to assign to this network interface.')] [string] $PrivateIpAddress,

[Parameter(Mandatory=$false, HelpMessage='Internal DNS name label for the new network interface')] [string] $InternalDnsNameLabel,

[Parameter(Mandatory=$false, HelpMessage='DNS server for the network interface')] [string[]] $DnsServer,

[Parameter(Mandatory=$false, HelpMessage='Key-value pairs in the form of a hash table. For example: @{key0="value0";key1=$null;key2="value2"}')] [hashtable] $Tag,

## New-AzLwResources

Parameters: <string> **Location**, <string> **ResourceGroupName**, <string> **StorageAccountName**, <string> SkuName, <hashtable[]> SecurityGroups, <hashtable[]> VirtualNetworks, <hashtable[]> IPAddresses, <hashtable[]> VMs

Location - region

ResourceGroupName - name of the resource group in which to create

StorageAccountName - name of the storage account

SkuName – one of 'Standard\_LRS', 'Standard\_ZRS', 'Standard\_GRS', 'Standard\_RAGRS', 'Premium\_LRS'

SecurityGroups - array with Security Groups settings.

VirtualNetworks - array with VirtualNetworks settings.

IPAddresses - array with IPAddresses settings.

VMs - array with VMs settings.

For arrays with settings ResourceGroupName, Location, StorageAccountName are taken from generic settings above.