Archive Shuttle 11.5

PowerShell Guide

Quest Software Inc.
Attn: LEGAL Dept.
20 Enterprise, Suite 100
Aliso Viejo, CA 92656

Refer to our Web site (https://www.quest.com) for regional and international office information.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at https://www.quest.com/legal.

Trademarks

Quest, the Quest logo, and Metalogix are trademarks and registered trademarks of Quest Software Inc. and its affiliates. For a complete list of Quest marks, visit https://www.quest.com/legal/trademark-information.aspx. All other trademarks and registered trademarks are the property of their respective owners.

Legend

CAUTION: A caution icon indicates potential damage to hardware or loss of data if instructions are not followed.

IMPORTANT, NOTE, TIP, MOBILE OR VIDEO: An information icon indicates supporting information.

Archive Shuttle
Updated October 2024
Version 11.5

Contents

1

Archive Shuttle and PowerShell

The PowerShell module allows you to manage Archive Shuttle migrations using a command line interface approach. Using PowerShell means that much of the administration work can be configured to be executed remotely, and it can be part of other scripted solutions (such as AD user migration, or Exchange Mailbox migration).

PowerShell module requirements

Before installing the PowerShell Module the following requirements need to be met:

- .Net Framework 4.0
- Exchange Online PowerShell module
  - Minimum: v3.0.0
  - Recommended: v3.2.0
  - Support up to: v3.5.1

NOTE: PowerShell compliance commands still need to use Global Admin credentials to connect to Exchange Online. Microsoft does not support AccessToken parameter for the Connect-IPPSSession command. PowerShell compliance is used to get a list of compliance tags.

Connecting to Archive Shuttle Core

To issue any commands via PowerShell, the first step is to connect to the Archive Shuttle Core. This can be done as follows:

```
$Core = Connect-ASCore –ComputerName “Name of Server” –Credential “some domain\user”
```

You will then be prompted for the password for the account.

The $Core variable can then be used for additional commands, as shown in the example in the next section.

To connect to a Cloud Core, use:

```
$Core = Connect-ASCore –ComputerName cloudtenantname.archiveshuttle.com -AuthenticationType Basic -UseHTTPS –Credential “cloud account”
```

NOTE: PowerShell should be launched as an administrator.

Performing a Command

Once a connection has been made to the Archive Shuttle Core, commands can be executed. For example, a list of current mappings can be shown as follows:

```
Get-ASMappings –Core $Core
```

PowerShell paging functionality

The standard PowerShell paging functionality commands can be used with the Archive Shuttle cmdlets. For example:

```
Get-ASContainers -Core $core -first 10 -skip 0
```

2

Get Archive Shuttle commands

Begin by running PowerShell as an Administrator on your virtual machine. The following PowerShell commands can be performed:

Get-Module -ListAvailable	/* list of all installed PS modules – should contain Archive Shuttle PowerShell version. 1.0 */
Get-Command -Module ArchiveShuttle.PowerShell	/* list of all public Archive Shuttle commands for PowerShell*/
Get-Command -Module ArchiveShuttle.PowerShell*|Sort-Object ModuleName,Name	/* list of all Archive Shuttle commands (public and extended) for PowerShell*/

3

Public commands table

This table contains list of all PowerShell commands with version where command was created or updated.

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Add-ASContainer	X	7.7	9.7
Add-ASContainerMapping		<= 7.0	10.5
Add-ASContainerToTag		7.7	9.7
Add-ASContainerToUser		10.5	
Add-ASEnvironment		<= 7.0	9.7
Add-ASEVRetentionCategoryMapping		8.3	9.8
Add-ASFileNamePolicy		8.3	
Add-ASFilterCondition		7.9	
Add-ASFilterPolicy		7.9	
Add-ASFolderNamePolicy		8.4	
Add-ASGroup		7.5	7.7
Add-ASItemDatabase		<= 7.0	9.0
Add-ASJEADUserRule		9.4	9.7

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Add-ASJEDistributionListRule		10.1	
Add-ASJEEmailAddressRule		9.4	9.7
Add-ASJELeaverMapping		9.4	10.0
Add-ASJEUserGroup		8.4	9.7
Add-ASJEUserMapping		8.4	10.0
Add-ASLink		<= 7.0	10.1
Add-ASMappingConfigurationTemplate		9.3	
Add-ASModulePool		N/A	
Add-ASModuleToModulePool		N/A	
Add-ASO365LeaversConfiguration		8.3	10.6
Add-ASPstSourcePath		<= 7.0	
Add-ASSettingDefinitionToSchedule	Removed	8.3	9.1
Add-ASSettingSchedule		8.3	9.1
Add-ASTag		7.7	
Add-ASTargetPathNameP		10.2	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
olicy			
Add-ASUserToGroup		7.5	7.6
Add-ASWorkflowPolicy		7.9	10.5
Add-ASWorkflowPolicyStep		7.10	9.0
Connect-ASCore	X	6.7	7.7
Copy-ASJEAutoCreationRules		9.4	10.1
Disable-ASActiveDirectorySync		<= 7.0	
Disable-ASArchiveGathering		<= 7.0	
Disable-ASCollection		<= 7.0	
Disable-ASMigration		<= 7.0	
Disable-ASModule		<= 7.0	
Enable-ASActiveDirectorySync		<= 7.0	
Enable-ASArchiveGathering		<= 7.0	
Enable-ASCollection		<= 7.0	
Enable-ASMigration		<= 7.0	
Enable-ASModule		<= 7.0	
Enable-ASStage2		<= 7.0	8.2
Get-AdamContainer		7.7	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Get-ASADDomain		7.7	
Get-ASADGroup		7.7	
Get-ASADGroupMember		7.7	
Get-ASArchive		10.1	
Get-ASBulkMappingWizardTemplate		10.6	
Get-ASCommandQueueStatus	X	7.7	
Get-ASConfiguration	X	<= 7.0	9.1
Get-ASContainers		<= 7.0	9.7
Get-ASCoreVersion		8.0	
Get-ASEnvironment		7.8	9.7
Get-ASEVRetentionCategoryMapping		8.3	
Get-ASExchangeDatabase		7.7	
Get-ASExchangeServer		7.7	
Get-ASFailedItemThreshold		7.7	
Get-ASFileNamePolicy		8.3	
Get-ASFilterPolicy		<= 7.0	7.2
Get-ASFolderNamePolicy		8.4	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Get-ASGroup		7.5	7.7
Get-ASHealthStatus		10.5	
Get-ASJEADUserRule		9.4	10.1
Get-ASJEDistributionListRule		10.1	
Get-ASJEEmailAddressRule		9.4	10.1
Get-ASJELeaverMapping		9.4	9.7
Get-ASJELeaversStatus		9.6	
Get-ASMailboxQuotaStatus		10.7	
~~Get-ASJEMailboxQuotaStatus~~		9.5	10.7 (removed)
Get-ASJESenderRecipient		8.4	10.0
Get-ASJEStage1Statistics		8.4	
Get-ASJEUserGroup		8.4	9.7
Get-ASJEUserMapping		8.4	9.3
Get-ASLicense		10.1	
Get-ASLink		<= 7.0	9.7
Get-ASLinkHealth		7.7	9.7
Get-ASLinkModule		<= 7.0	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Get-ASMailbox		7.7	10.1
Get-ASMappingConfigurationTemplate		9.3	
Get-ASMappings		<= 7.0	10.0
Get-ASModule		<= 7.0	9.9
Get-ASModuleCredential		10.7	
Get-ASModulePerformance		7.9	7.10
Get-ASModulePool		N/A	
Get-ASO365LeaversConfiguration		8.3	9.4
Get-ASO365License		8.3	
Get-ASPremigrationStatus		10.7	
Get-ASProgressStatistics		10.1	
Get-ASRetentionCategory		7.9	7.10
Get-ASScheduledTask	X	10.1	
Get-ASSettingSchedule		8.3	9.1
Get-ASStage1Statistics		<= 7.0	10.0
Get-ASStage2Status		<= 7.0	10.0

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Get-ASTag		7.7	
Get-ASTargetPathNamePolicy		10.2	
Get-ASUser		7.7	9.7
Get-ASUserSyncStatus	X	7.5	
Get-ASVaultStore		<= 7.0	
Get-ASWatermark		7.7	8.1
Get-ASWorkflowPolicy		<= 7.0	7.2
Get-ASWorkflowPolicyStep		7.10	8.2
Recreate-ASMapiProfile		9.8	
Remove-ASContainerMapping		7.10	8.0
Remove-ASJEADUserRule		9.4	10.1
Remove-ASJEDistributionListRule		10.1	
Remove-ASJEEmailAddressRule		9.4	10.1
Remove-ASJELeaverMapping		9.4	9.7
Remove-ASJEUserMapping		8.4	9.7

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Remove-ASLinkDatabase	X	<= 7.0	9.0
Remove-ASMappingConfigurationTemplate		9.3	
Remove-ASModuleFromModulePool		N/A	
Remove-ASModulePool		N/A	
Remove-ASPstSourcePath		<= 7.0	
Restart-ASModule		7.8	
Set-ASCommandIntervalLastExecutedDate	X	7.7	
Set-ASConfiguration	O	<= 7.0	9.7
Set-ASContainer		10.4	
Set-ASContainerMapping		7.8	10.5
Set-ASEnvironment		<= 7.0	9.7
Set-ASFailedItemThreshold		7.7	
Set-ASFileNamePolicy		8.3	
Set-ASFolderNamePolicy		8.4	
Set-ASGroup		7.5	7.7

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Set-ASJEDistributionListRule		10.1	
Set-ASJEEmailAddressRule		9.4	10.1
Set-ASJELeaverMapping		9.4	9.7
Set-ASJELeaversStatus		9.6	9.9
Set-ASJEUserGroup		8.4	9.7
Set-ASJEUserMapping		8.4	9.7
Set-ASLinkModule		<= 7.0	7.8
Set-ASMappingConfigurationTemplate		9.3	
Set-ASModule		<= 7.0	9.7
Set-ASModuleCredential		10.7	
Set-ASModuleLogLevel		<= 7.0	
Set-ASModulePool		N/A	
Set-ASO365LeaversConfiguration		8.3	10.6
Set-ASPremigrationAction		10.5	
Set-ASPstSourcePath		<= 7.0	
Set-ASPstTargetPath		<= 7.0	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Set-ASPstTemporaryPath		9.8	
Set-ASScheduledTask	X	10.1	
Set-ASSettingSchedule		8.3	
Set-ASStage2Status		9.9	
Set-ASStagingAreaPath		<= 7.0	
Set-ASTag		7.7	
Set-ASTargetPathNamePolicy		10.2	
Set-ASWatermark		7.7	8.1
Set-ASWorkflowPolicy		7.9	7.10
Set-ASWorkflowPolicyStep		7.10	9.0
Start-ASADSyncSingleUser	X	7.5	
Start-ASArchiveGathering		<= 7.0	
Start-ASCommand	X	<= 7.0	
Start-ASModule		<= 7.0	9.7
Start-ASO365SyncMailboxes		<= 7.0	
Start-ASO365SyncSingleUser	X	7.5	

Name	PowerShell specific (N/A in UI)	Created in (version)	Updated in (version)
Start-ASRetryFailedItems		7.8	
Start-ASScheduledTask	Removed	<= 7.0	10.1
Start-ASStagingAreaCleanup		9.2	
Start-ASSyncADDomains		<= 7.0	
Start-ASSyncADUsers		<= 7.0	
Stop-ASModule		<= 7.0	
Update-ASMailboxSizeAndQuota		10.7	

4

Available commands sorted to groups

Core

- Connect-ASCore
- Connect to core in AS.CLOUD

Configuration

- Get-ASConfiguration
- Set-ASConfiguration
- Add-ASSettingSchedule
- Set-ASSettingSchedule
- Get-ASSettingSchedule
- Add-ASSettingDefinitionToSchedule
- Get-ASFailedItemThreshold
- Set-ASFailedItemThreshold
- Get-ASWatermark
- Set-ASWatermark
- Get-ASRetentionCategory
- Get-ASEVRetentionCategoryMapping
- Add-ASEVRetentionCategoryMapping
- Add-ASWorkflowPolicy
- Set-ASWorkflowPolicy
- Get-ASWorkflowPolicyStep
- Add-ASWorkflowPolicyStep
- Set-ASWorkflowPolicyStep
- Add-ASFilterCondition

- Add-ASFilterPolicy
- Get-ASFileNamePolicy
- Set-ASFileNamePolicy
- Add-ASFileNamePolicy
- Get-ASTargetPathNamePolicy
- Set-ASTargetPathNamePolicy
- Add-ASTargetPathNamePolicy
- Get-ASO365License
- Add-ASO365LeaversConfiguration
- Set-ASO365LeaversConfiguration
- Get-ASO365LeaversConfiguration

Module

- Get-ASModule
- Get-ASModuleCredential
- Set-ASModule
- Set-ASModuleCredential
- Set-ASModuleLogLevel
- Get-ASModulePerformance
- Enable-ASModule
- Disable-ASModule
- Start-ASModule
- Stop-ASModule
- Restart-ASModule

Command

- Start-ASCommand
- Get-ASCommandQueueStatus
- Set-ASCommandIntervalLastExecutedDate
- Recreate-ASMapiProfile

Scheduled Task

- Get-ASScheduledTask
- Set-ASScheduledTask

Active Directory

- Enable-ASActiveDirectorySync
- Disable-ASActiveDirectorySync
- Get-ASADGroup
- Get-ASADGroupMember

Domain

- Get-ASADDomain
- Start-ASSyncADDomains

Exchange

- Get-ASExchangeServer
- Get-ASExchangeDatabase

Environment

- Add-ASEnvironment
- Set-ASEnvironment
- Get-ASEnvironment

Vault Store

- Get-ASVaultStore

Archive Gathering

- Enable-ASArchiveGathering
- Disable-ASArchiveGathering
- Start-ASArchiveGathering

License

- Get-ASLicense

Link

- Add-ASLink
- Get-ASLink
- Get-ASLinkHealth
- Get-ASLinkModule
- Set-ASLinkModule
- Remove-ASLinkDatabase
- Add-ASItemDatabase

Staging Area

- Set-ASStagingAreaPath
- Start-ASStagingAreaCleanup

Performance

- Get-ProgressStatistics
- Get-ASHealthStatus

PST

- Add-ASPstSourcePath
- Remove-ASPstSourcePath
- Set-ASPstSourcePath
- Set-ASPstTargetPath
- Set-ASPstTemporaryPath

User & Container & Mailbox

- Add-ASContainer
- Add-ASContainerToUser
- Get-ASArchive

- Get-ASUser
- Get-ASContainers
- Get-AdamContainer
- Get-ASMailbox
- Get-ASUserSyncStatus
- Get-ASMailboxQuotaStatus
- Update-ASMailboxSizeAndQuota
- Set-ASContainer
- Start-ASSyncADUsers
- Start-ASO365SyncMailboxes
- Start-ASADSyncSingleUser
- Start-ASO365SyncSingleUser

Mapping

- Add-ASContainerMapping
- Set-ASContainerMapping
- Get-ASBulkMappingWizardTemplate
- Get-ASMappings
- Get-ASPremigrationStatus
- Get-ASStage1Statistics
- Get-ASStage2Status
- Get-ASWorkflowPolicy
- Get-ASFilterPolicy
- Enable-ASCollection
- Enable-ASMigration
- Disable-ASCollection
- Disable-ASMigration
- Enable-ASStage2
- Start-ASRetryFailedItems
- Remove-ASContainerMapping
- Add-ASMappingConfigurationTemplate

- Set-ASMappingConfigurationTemplate
- Get-ASMappingConfigurationTemplate
- Remove-ASMappingConfigurationTemplate
- Set-ASStage2Status
- Set-ASPremigrationAction

Journal Explosion

- Add-ASJEUserGroup
- Get-ASJEUserGroup
- Set-ASJEUserGroup
- Add-ASFolderNamePolicy
- Set-ASFolderNamePolicy
- Get-ASFolderNamePolicy
- Get-ASJESenderRecipient
- Add-ASJEUserMapping
- Get-ASJEUserMapping
- Set-ASJEUserMapping
- Remove-ASJEUserMapping
- Add-ASJELeaverMapping
- Get-ASJELeaverMapping
- Set-ASJELeaverMapping
- Remove-ASJELeaverMapping
- Get-ASJEStage1Statistics
- Add-ASJEEmailAddressRule
- Get-ASJEEmailAddressRule
- Set-ASJEEmailAddressRule
- Remove-ASJEEmailAddressRule
- Add-ASJEADUserRule
- Get-ASJEADUserRule
- Set-ASJEADUserRule
- Remove-ASJEADUserRule

- Add-ASJEDistributionListRule
- Get-ASJEDistributionListRule
- Remove-ASJEDistributionListRule
- Set-ASJEDistributionListRule
- Copy-ASJECreationRules
- Get-ASJELeaversStatus
- Set-ASJELeaversStatus

Groups & Tags

- Add-ASGroup
- Set-ASGroup
- Get-ASGroup
- Add-ASUserToGroup
- Add-ASTag
- Set-ASTag
- Get-ASTag
- Add-ASContainerToTag

5

Available commands

Connect-ASCore

Create core via Connect-ASCore command.

SYNOPSIS

Connects to Archive Shuttle core.

SYNTAX

Connect-ASCore [-ComputerName] <string> [-Credential] <PSCredential> -UseHttps [[-Port] <int>] [<CommonParameters>]

Connect-ASCore [-RemoteAddress] <string> [-Credential] <PSCredential> [-AuthenticationType <AuthenticationType>] [[-SendTimeout] <string>] [<CommonParameters>]

DESCRIPTION

Connects to Archive Shuttle core and creates PowerShellModule service instance which allows all commands to communicate with Archive Shuttle core. This command is used as an input parameter for all other Archive Shuttle PowerShell commands.

Credentials (mandatory) - insert the domain with the username, and after command is executed, the user will be prompted to add a password.

RemoteAddress - Uses specified remote service address. Full URL is required.

ComputerName - Connects to specified computer name.

UseHttps - Uses HTTPS protocol instead of HTTP.

Port - Specifies communication port number. By default, port 80 is used for HTTP and port 443 is used for HTTPS.

Examples of Connect-ASCore usage:

$core = Connect-ASCore -ComputerName "COMPUTER01" -Credential "domain\username"	/* This will connect to PowerShell service on server COMPUTER01 */
$core = Connect-ASCore -ComputerName "COMPUTER01" –Port 567 -Credential "domain\username"	/* This will connect to PowerShell service on server COMPUTER01 with port 567 */

$core = Connect-ASCore -Credential "domain\username"	/* This will connect to default server selected during modules configuration */

$password = convertto-securestring -String "EnterPasswordHere" -AsPlainText -Force

$UserCredential = New-object -typename System.Management.Automation.PSCredential -argumentlist "Domain\User",$password

Connect-ASCore -ComputerName "EnterComputerName" -Credential $UserCredential

Connect to core in AS Cloud

In the case that core is installed in the cloud, it is possible to connect from a machine where Archive Shuttle modules (PowerShell module) is installed:

$UserCredential = Get-Credential -Message Credentials -UserName user@quadrotech-it.com

Connect-ASCore -ComputerName cloud.archiveshuttle.com -AuthenticationType Basic -UseHttps -Credential $UserCredential

Core is a **mandatory** parameter for each command as it is used to get data from core.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. call Connect-ASCore once and core parameter will be automatically added from session state and other commands can be used without core from this point.

Sometimes command may time-out. Default time is seconds. This can be increased: *$core = Connect-ASCore $computerName -Credential $credentials -SendTimeout 120*

Set-ASConfiguration

SYNOPSIS

Sets configuration setting.

SYNTAX

Set-ASConfiguration [-Core] <ConnectCore> [-SettingDefinition] <SettingDefinitionEnum> [-Value] <object> [[-ModuleId] <Guid[]>] [[-SettingScheduleId] <int>] [[-ContainerMappingTemplateId] <int>] [<CommonParameters>]

DESCRIPTION

The Set-ASConfiguration cmdlet sets configuration setting. After the command is executed, the result can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASConfiguration -SettingDefinition EVCollectorParallelism -Value 5	/* Sets Enterprise Vault Collector module parallelism to 5 for global configuration */
Set-ASConfiguration -SettingDefinition MyTimezoneOverride -Value "(UTC) Coordinated Universal Time" -UserName "domain\user01"	/* Sets TimeZoneOverride setting for specified user in global configuration */
Set-ASConfiguration -SettingDefinition PstExportFileParallelism -Value 8 -ModuleId c9a0d554-1980-e811-838a-005056982247 Set-ASConfiguration -SettingDefinition PstExportFileParallelism -Value 8 -ModuleId c9a0d554-1980-e811-838a-005056982247 -SettingScheduleId 1	/* Sets PST Export paralelism of GLOBAL schedule for specified module */ /* Sets PST Export paralelism of specified custom schedule for specified module */
Set-ASConfiguration -SettingDefinition AutoRecreateLeavers 0 -ContainerMappingTemplateId 1 Set-ASConfiguration -SettingDefinition Office365VirtualJournalItemCountLimit 10 -ContainerMappingTemplateId 1 Set-ASConfiguration -SettingDefinition Office365VirtualJournalItemSizeByteLimit 1024000 -ContainerMappingTemplateId 1	/* Mapping template configuration details are defined through Set-ASConfiguration command. It works the same way as when defining general system configuration but with a special parameter - ContainerMappingTemplateId is specified */

Get-ASConfiguration

SYNOPSIS

Gets configuration.

SYNTAX

Get-ASConfiguration [-Core] <ConnectCore> [[-SettingDefinition] <SettingDefinitionEnum>] [[-SettingGroup] <SettingGroupEnum>] [[-UserName] <string>] [[-ModuleId] <Guid[]>] [-SettingScheduleId <int>] [<CommonParameters>]

DESCRIPTION

The Get-ASConfiguration cmdlet gets a list of configurations. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASConfiguration -SettingDefinition EVCollectorParallelism	/* Gets Enterprise Vault Collector module parallelism global setting */
Get-ASConfiguration -SettingDefinition MyTimezoneOverride -UserName "domain\username" | fl Get-ASConfiguration -SettingDefinition MyTimezoneOverride	/* Gets Timezone global setting of specified user */ /* Gets Timezone global setting of all users */
Get-ASConfiguration -SettingScheduleId 1 Get-ASConfiguration -SettingDefinition EVCollectorParallelism -SettingScheduleId 1 Get-ASConfiguration -ModuleId 96f342f8-4b7f-e811-838a-005056982247 Get-ASConfiguration -ModuleId c9a0d554-1980-e811-838a-005056982247 -SettingScheduleId 1	/* Gets all settings related to specified schedule profile */ /* Gets specified setting of schedule profile */ /* Gets settings of GLOBALschedule for specified module */ /* Gets settings of specific schedule for specified module */

Add-ASSettingSchedule

SYNOPSIS

Add Archive Shuttle setting schedule.

SYNTAX

Add-ASSettingSchedule [-Core] <ConnectCore> [-Name] [<string>] [-Schedule] [<string>] [-Color] [<string>] [-ModuleId] <Guid> [<CommonParameters>]

DESCRIPTION

The Add-ASSettingSchedule cmdlet creates new schedule for setting definition. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASSettingSchedule -Name SCHEDULE01 -Color "#FF00FF"	/* Creates custom schedule profile */
Add-ASSettingSchedule -Name SCH01 -Color "#FF00FF" -Schedule "3cf8000ff8000ff8000ff8000ff8000ff8000ff8000f3c" -ModuleId b44e3afe-4b7f-e811-838a-005056982247	/* Creates custom schedule profile with schedule assignment for specified module (command accepts only hexa format of schedule assignment) */

Add-ASSettingSchedule -Name SCH02 -Color "#FF00FF" -Schedule "3cf8000ff8000ff8000ff8000ff8000ff8000ff8000f3c" -ModuleId ((Get-ASModule -ModuleType EVExport -ComputerName Computer01).ModuleId)	

Set-ASSettingSchedule

SYNOPSIS

Set Archive Shuttle setting schedule.

SYNTAX

Set-ASSettingSchedule [-Core] <ConnectCore> [-Id] [<int>] [-Name] [<string>] -Schedule] [<string>-Color] [<string> [<CommonParameters>]

DESCRIPTION

The Set-ASSettingSchedule cmdlet sets schedule name, color or schedule hours. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASSettingSchedule -Id 1 -Color "#FF00FF" -Schedule "3c020000020000020000020000020000020000020003c"	/* Sets color and schedule of specified existing schedule */
Get-ASSettingSchedule -SettingScheduleId 1 | Set-ASSettingSchedule -Name "CHANGEDNAME"	/* Gets specified schedule and changes it's name */

Get-ASSettingSchedule

SYNOPSIS

Get Archive Shuttle setting schedule.

SYNTAX

Get-ASSettingSchedule [-Core] <ConnectCore> [-SettingScheduleId] [<int[]>] [-Name] [<string[]>] [[-ModuleId] <Guid>] [<CommonParameters>]

DESCRIPTION

The Get-ASSettingSchedule cmdlet gets schedule for setting definition. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASSettingSchedule | Format-Table -AutoSize	/* Gets all available custom schedule profiles */
Get-ASSettingSchedule -Name SCHEDULE01	/* Gets specified custom schedule based on schedule Name */
Get-ASSettingSchedule -ModuleId c9a0d554-1980-e811-838a-005056982247	/* Gets custom schedules of specified module */

Add-ASSettingDefinitionToSchedule

SYNOPSIS

Add Archive Shuttle setting definition into schedule.

SYNTAX

Add-ASSettingDefinitionToSchedule [-Core] <ConnectCore> [-SettingDefinitionId] [<int>] [-SettingSchedulerId] [<SettingSchedulerId>] -DefaultValueNumeric] [<long>-DefaultValueText] [<string> [<CommonParameters>]

DESCRIPTION

The Add-ASSettingDefinitionToSchedule cmdlet creates relation between schedule and setting definition. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASSettingDefinitionToSchedule -SettingSchedulerId 1 -SettingDefinition EVCollectorParallelism -DefaultValueNumeric 12	/* Adds custom configuration parameter into specified schedule profile

Get-ASFailedItemThreshold

SYNOPSIS

Gets failed item threshold.

SYNTAX

Get-ASFailedItemThreshold [-Core] <ConnectCore> [-LinkId] <Guid[]> [-ContainerMappingId] <int[]> [-GetAll] <SetFailedItemThresholdTable> [<CommonParameters>]

DESCRIPTION

The Get-ASFailedItemThreshold cmdlet returns value of Failed item threshold. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASFailedItemThreshold -GetAll Link Get-ASFailedItemThreshold -GetAll ContainerMapping	/* Retrieves threshold for all links or all containers */
Get-ASFailedItemThreshold -LinkId "5bb2fc14-5cc1-461d-951c-ff0618b27883"	/* Retrieves threshold for specified link only */
Get-ASLink -Type EnterpriseVault | Get-ASFailedItemThreshold	/* Retrieves threshold for Enterprise Vault links only */
Get-ASFailedItemThreshold -GetAll Link | Where-Object {$_.FailedItemThreshold -eq 2}	/* Retrieves all links with threshold=2 */

Set-ASFailedItemThreshold

SYNOPSIS

Sets failed item threshold.

SYNTAX

Set-ASFailedItemThreshold [-Core] <ConnectCore> [-LinkId] <Guid[]> [-ContainerMappingId] <int[]> [-SetAll] <SetFailedItemThresholdTable> [-FailedItemThreshold] <int[]> [<CommonParameters>]

DESCRIPTION

The Set-ASFailedItemThreshold cmdlet Sets value for Failed item threshold. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

There are three variations of parameters:

1. LinkId, FailedItemThreshold
2. ContainerMappingId, FailedItemThreshold
3. SetAll(None/Link/ContainerMapping), FailedItemThreshold

Example:

Set-ASFailedItemThreshold -FailedItemThreshold 1 -LinkId "83139672-f03c-e611-813c-005056b84ed8"	/* Sets threshold for specified link only */
Set-ASFailedItemThreshold -FailedItemThreshold 1 -ContainerMappingId 1	/* Sets threshold for specified mapping only (it's possible to enter multiple mappingIds separated by comma) */
Set-ASFailedItemThreshold -FailedItemThreshold 1 -SetAll ContainerMapping	/* Sets threshold for all existing mappings (can be also set for all existing links) */
Get-ASLink -Type EnterpriseVault | Set-ASFailedItemThreshold -FailedItemThreshold 1	/* Sets threshold for all EV links only */

Get-ASWatermark

SYNOPSIS

Gets low or high watermark of link path.

SYNTAX

Get-ASWatermark [-Core] <ConnectCore> [-LinkId] <Guid[]> [[-Type] <WatermarkType>] [<CommonParameters>]

DESCRIPTION

The Get-ASWatermark cmdlet gets value for low or high watermark of links staging area path (-Type StagingAreaPath) or PST output path (-Type PstOutputPath). If no LinkId is set, it will return staging area watermark values of all links. If no Type is specified, StagingAreaPath type will be used as default. After the command is executed, the module status can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Staging Area:

LowWaterMark - If the free disk space on a links' staging area drops below this value, exporting of new data will be suspended.

HighWaterMark - If the amount of data which has been exported, but not imported for a particular link exceeds this value, exporting of new data will be suspended.

PST Output Path:

LowWatermark: If the free disk space where the PST output path folder is located drops below this value, moving of PST files from staging area will be suspended.

HighWatermark: If the amount of data within the PST output path exceeds this value, moving of PST files from staging area will be suspended.

Example:

Get-ASWatermark Get-ASWatermark -LinkId "f3872fec-d9dd-459d-8e2b-c68156a9d81e" -Type StagingAreaPath -LowWatermark 1474836480 -HighWatermark 1474836480 Get-ASWatermark -LinkId "f3872fec-d9dd-459d-8e2b-c68156a9d81e" -Type PstOutputPath -LowWatermark 1474836480 -HighWatermark 1474836480	/* Retrieves staging area watermarks for all links*/ /* Retrieves staging area watermarks of specified link*/ /* Retrieves PST output path watermarks of specified PST link*/
Get-ASWatermark -LinkId "f3872fec-d9dd-459d-8e2b-c68156a9d81e"	/* Retrieves staging area watermarks for specified link */
Get-ASLink -LinkName "EnterpriseVault10" | Get-ASWatermark	/* Retrieves staging area watermarks of link with specified name */
Get-ASLink -Type EnterpriseVault | Get-ASWatermark	/* Retrieves staging area watermarks for all Enterprise Vault links */

Set-ASWatermark

SYNOPSIS

Sets low or high watermark of link path.

SYNTAX

Set-ASWatermark [-Core] <ConnectCore> [-LinkId] <Guid[]> [-LowWatermark] <long[]> [-HighWatermark] <long[]> [[-Type] <WatermarkType(StagingAreaPath/PstOutputPath)>] [<CommonParameters>]

DESCRIPTION

The Set-ASWatermark cmdlet sets value for low or high watermark of links staging area path (-Type StagingAreaPath) or PST output path (-Type PstOutputPath). If no LinkId is set, it will set staging area watermark values of all links. If no Type is specified, StagingAreaPath type will be used as default. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Staging Area:

LowWaterMark - If the free disk space on a links' staging area drops below this value, exporting of new data will be suspended.

HighWaterMark - If the amount of data which has been exported, but not imported for a particular link exceeds this value, exporting of new data will be suspended.

PST Output Path:

LowWatermark: If the free disk space where the PST output path folder is located drops below this value, moving of PST files from staging area will be suspended.

HighWatermark: If the amount of data within the PST output path exceeds this value, moving of PST files from staging area will be suspended.

Example:

Set-ASWatermark -LowWatermark 1474836480 -HighWatermark 1474836480 Set-ASWatermark -Type StagingAreaPath -LowWatermark 1474836480 -HighWatermark 1474836480 Set-ASWatermark -Type PstOutputPath -LowWatermark 1474836480 -HighWatermark 1474836480	/* Sets watermark 1 GB for all staging area paths existing in Archive Shuttle system */ /* Sets watermark 1 GB for all staging area paths existing in Archive Shuttle system */ /* Sets watermark 1 GB for all PST Output paths of PST links existing in Archive Shuttle system */
Set-ASWatermark -LinkId "e4b4d6e1-10e4-4bdd-8adb-ff8605b81538" -LowWatermark 5474836480 -HighWatermark 5474836480	/* Sets watermark for staging area related to specified link */
Get-ASLink -LinkName "EnterpriseVault10" | Set-ASWatermark -LowWatermark 15474836480 -HighWatermark 5474836480	/* Sets watermark for staging area related to specified link */

Get-ASRetentionCategory

SYNOPSIS

Gets retention category.

SYNTAX

Get-ASRetentionCategory [[-Core] <ConnectCore>] [[-Name] <string[]>] [[-EVSiteId] <string[]>] [[-EVRetentionCategoryId] <string[]>] [<CommonParameters>]

DESCRIPTION

The Get-EVRetentionCategory return EV retention category. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASRetentionCategory	/* Retrieves all EV Retention Categories */
Get-ASRetentionCategory -EVRetentionCategoryId 173E142CD3FDE11479691B286FF8B920B1b10000QA-EV10-01.qa.lab.quadrotech-it.com	/* Retrieves specified EV Retention Category based on Retention Category ID*/
Get-ASRetentionCategory | Where-Object { $_.Name -like "TEST*"}| Format-Table -AutoSize	/* Retrieves all EV Retention Categories with name starting with "TEST" + results are formatted in table */

Get-ASEVRetentionCategoryMapping

SYNOPSIS

Get retention category mapping.

SYNTAX

Get-ASEVRetentionCategoryMapping [-Core] <ConnectCore> -SourceSiteName] [<string>-SourceSiteId] [<string>-SourceRetentionCategoryName] <string>] [[-SourceRetentionCategoryId] <string>] [[-TartgetSiteName] [<string>-TartgetSiteId] [<string>

-TartgetRetentionCategoryName] [<string>-TartgetRetentionCategoryId] [<string> [<CommonParameters>]

DESCRIPTION

The Get-ASEVRetentionCategoryMapping cmdlet gets retention category mapping. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASEVRetentionCategoryMapping -IncludeTotalCount	/* Retrieves all Enterprise Vault retention category mappings and total count of results */
Get-ASEVRetentionCategoryMapping -SourceSiteName "EV12" | Format-Table -AutoSize	/* Retrieves only Enterprise Vault retention category mappings of specified Enterprise Vault site */

Add-ASEVRetentionCategoryMapping

SYNOPSIS

Add retention category mapping.

SYNTAX

Add-ASEVRetentionCategoryMapping [-Core] <ConnectCore> [-Intrasite] <SwitchParameter> [-SourceEVRetentionCategoryId] <SwitchParameter> [-SourceEVSite] <string> [-SourceRetentionCategoryName] <string>[-TargetEVRetentionCategoryId] <string> [-TargetO365RetentionId] <string>[[-TargetEVSite] <string>] [[-TargetO365LinkId] <string>] [-TargetRetentionName] <string> [<CommonParameters>]

DESCRIPTION

The Add-ASEVRetentionCategoryMapping cmdlet creates new retention category mapping. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASEVRetentionCategoryMapping -Intrasite	/* Creates mapping for all intrasite retention categories (UI action button "Add Intrasite Migration Mappings") */
Add-ASEVRetentionCategoryMapping -SourceEVSite "EV10" -SourceRetentionCategoryName "Name1" TargetEVSite "EV12" -TargetRetentionName "Name2"	/* Creates retention maping with specified parameters for EV Extrasite migrations */
Add-ASEVRetentionCategoryMapping -SourceEVSite "EV10"-SourceRetentionCategoryName "Name1" -TargetO365 "O365GUID" -TargetRetentionName "O365TAG"	/* Creates retention maping with specified parameters for EV to O365 migrations - feature message stamping with O365 retention tag*/

Add-ASWorkflowPolicy

SYNOPSIS

Add Archive Shuttle workflow policy.

SYNTAX

Add-ASWorkflowPolicy [-Core] <ConnectCore> [-WorkflowPolicyName] <string> [-Description] <string> [-ToContainerTypes] <WorkflowPolicyContainerTypeEnum[]> [[-IsSystem]

<SwitchParameter>] [[-Enabled] <bool>] [[-WorkflowPolicyType] <StageType>] [<CommonParameters>]

DESCRIPTION

The Add-ASWorkflowPolicy cmdlet creates new workflow policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASWorkflowPolicy -WorkflowPolicyType Stage2 -WorkflowPolicyName WFPNAME01 -Description WFPDESC -ToContainerTypes Exchange,PST -Enabled 1	/* Creates new stage 2 workflow policy with specified parameters and is enabled */
Add-ASWorkflowPolicy -WorkflowPolicyType Stage2 -WorkflowPolicyName 03TEST -Description TESTDESC03 -IsSystem -Enabled 0	/* Creates new system stage 2 workflow policy (system=cannot be deleted) with specified parameters and is disabled */
Add-ASWorkflowPolicy -WorkflowPolicyType Premigration -WorkflowPolicyName 01PreMigrationWFPolicy -Description TESTPreMigWFPolicy -Enabled 1	/* Creates new pre-migration workflow policy with specified parameters and enabled*/

NOTE: This command only creates workflow policy without steps. These must be added via a different command.

Set-ASWorkflowPolicy

SYNOPSIS

Set AS workflow policy.

SYNTAX

Set-ASWorkflowPolicy [-Core] <ConnectCore> [-WorkflowPolicyId] [<int>] -WorkflowPolicyName] > [-Description] [<string>] [-ToContainerTypes] [<WorkflowPolicyContainerTypeEnum[]>] >-IsEnabled] [<bool> [<CommonParameters>]

DESCRIPTION

The Set-ASWorkflowPolicy cmdlet updates existing workflow policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASWorkflowPolicy -WorkflowPolicyId 1 -WorkflowPolicyName NEWNAME -Description NEWDESC -Enabled 0 -IsSystem 0 -ToContainerTypes None	/* Properties of the specified Workflow Policy are either replaced or changed
Get-ASWorkflowPolicy -WokflowPolicyId 1 | Set-ASWorkflowPolicy -Enabled 0	/* Specified Workflow Policy is retrieved and then set to disabled */

Get-ASWorkflowPolicyStep

SYNOPSIS

Get step of workflow policy.

SYNTAX

Get-ASWorkflowPolicyStep [-Core] <ConnectCore> [-WorkflowSequenceId] [<int>] [-WorkflowPolicyId] [<int>] [-CommandId] [<CommandEnum>] [<CommonParameters>]

DESCRIPTION

The Get-ASWorkflowPolicyStep cmdlet gets workflow policy step. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASWorkflowPolicyStep -WorkflowPolicyId 19 Get-ASWorkflowPolicyStep -WorkflowSequenceId 277 Result: WorkflowSequenceId : 277WorkflowPolicyId : 52 Command : Office365RemoveLicense CommandId : 1112 SequenceOrder : 1 WorkflowTriggerDefinition : NoTrigger IsWaitOnAllPrevious : False RunInParallelWithPrevious : False RetryFailed : True	/* Retrieves all steps related to specified Wotkflow Policy */ /* Retrieves details of exact step based on unique WorkflowSequenceId */

RetryFailedAfterMinutes : 5 RetryUnresponsive : True RetryUnresponsiveAfterMinutes : 5 IsContainerGranularityOnly : False	

Get-ASWorkflowPolicyStep -Command Office365RemoveLicense
 	
Get-ASWorkflowPolicyStep -Command Office365RemoveLicense | ForEach-Object {
$step = $_
$wfpolicy = Get-ASWorkflowPolicy -WorkflowPolicyId $step.WorkflowPolicyId
New-Object -TypeName PSObject -Property @{
WorkflowPolicyId = $step.WorkflowPolicyId
WorkflowName = $wfpolicy.WorkflowPolicyName
WorkflowSequenceId = $step.WorkflowSequenceId
Command = $step.Command
}
} | Select-Object -Property WorkflowName,WorkflowPolicyId,WorkflowSequenceId,Command |Sort-Object WorkflowName
	/* Retrieves all steps related to specified command with full details (retrieves list of all Workflow Policies where specified command is used) */

Get-ASWorkflowPolicy -First 2 | Get-ASWorkflowPolicyStep | Sort-Object WorkflowPolicyId,SequenceOrder | Format-Table -AutoSize	/* Retrieves Workflow Policy steps for all Workflow Policies */

Add-ASWorkflowPolicyStep

SYNOPSIS

Add command to workflow policy.

SYNTAX

Add-ASWorkflowPolicyStep [-Core] <ConnectCore> [-WorkflowPolicyId] <int> [-Command] <CommandEnum> -WorkflowTriggerDefinition] [<WorkflowTriggerDefinitionEnum[]>-IsWaitOnAllPrevious] [<SwitchParameter>-RunInParallelWithPrevious] [<SwitchParameter>-RetryFailed] [<SwitchParameter>-RetryFailedAfterMinutes] [<int>-RetryUnresponsive] [<SwitchParameter>-RetryUnresponsiveAfterMinutes] [<int>-IsContainerGranularityOnly] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Add-ASWorkflowPolicyStep cmdlet creates new workflow sequence = adds command to workflow policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASWorkflowPolicyStep -WorkflowPolicyId 50 -Command CollectItemsForArchive	/* New Workflow step is added into specified Workflow Policy (step is set to default settings) */
$newWfPolicy = Add-ASWorkflowPolicy -WorkflowPolicyName NewWfPolicy -Description "DescriptionDescription" -ToContainerTypes Office365 -Enabled 1 Add-ASWorkflowPolicyStep -WorkflowPolicyId $newWfPolicy.Id -Command WaitForImportFinished -WorkflowTriggerDefinition ManualSwitchOver -RetryFailed -RetryFailedAfterMinutes 5 -RetryUnresponsive -RetryUnresponsiveAfterMinutes 5 -IsWaitOnAllPrevious -RunInParallelWithPrevious -IsContainerGranularityOnly Add-ASWorkflowPolicyStep -WorkflowPolicyId $newWfPolicy.Id -Command Office365RemoveLicense -WorkflowTriggerDefinition ManualSwitchOver -RetryFailed -RetryFailedAfterMinutes 5 -RetryUnresponsive -RetryUnresponsiveAfterMinutes 5 -IsWaitOnAllPrevious -RunInParallelWithPrevious -IsContainerGranularityOnly	/* New Workflow Policy is created, then two new Workflow Steps are are added with specified settings (order of steps is dependent on commands sequence execution; order can be changed by other command afterwards) */

Set-ASWorkflowPolicyStep

SYNOPSIS

Set command to workflow policy.

SYNTAX

Set-ASWorkflowPolicyStep [-Core] <ConnectCore> [-WorkflowSequenceId] <int> -WorkflowPolicyId] [<int>-CommandId] <int>] [[-SequenceOrder] <int>] [[-WorkflowTriggerDefinition] [<WorkflowTriggerDefinitionEnum[]>-IsWaitOnAllPrevious] [<bool>

-RunInParallelWithPrevious] [<bool>-RetryFailed] [<bool>-RetryFailedAfterMinutes] [<int>-RetryUnresponsive] [<bool>-RetryUnresponsiveAfterMinutes] [<int>-IsContainerGranularityOnly] [<bool> [<CommonParameters>]

DESCRIPTION

The Set-ASWorkflowPolicyStep cmdlet updates workflow policy step = updates step of workflow policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASWorkflowPolicyStep -WorkflowPolicyId 48 -WorkflowSequenceId 265 -SequenceOrder 0	/* Changes order of specified Workflow Step within defined Workflow Policy */
Set-ASWorkflowPolicyStep -WorkflowPolicyId 48 -WorkflowSequenceId 265 -WorkflowTriggerDefinition NoTrigger -RunInParallelWithPrevious 0 -RetryFailed 0 -RetryFailedAfterMinutes 10	/* Changes various settings of specified Workflow Step within defined Workflow Policy */
Get-ASWorkflowPolicyStep -WorkflowPolicyId 48 | Set-ASWorkflowPolicyStep -WorkflowTriggerDefinition NoTrigger -RetryFailed 1 -RetryFailedAfterMinutes 120	/* Retrieves all Workflow Steps of specified Workflow Policy and then sets various settings for each */

Add-ASFilterPolicy

SYNOPSIS

Add new filter policies.

SYNTAX

Add-ASFilterPolicy [[-Core] <ConnectCore>] [[-Name] <string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASFilterPolicy cmdlet adds new Filter Policies. The Filter Policy can be used as an input parameter for the Add-ASContainerMapping command. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASFilterPolicy -Name "NEW FILTER POLICY NAME	/* Creates new Filter Policy (filter condition is required to be created afterwards) */

Add-ASFilterCondition

SYNOPSIS

Add new filter condition.

SYNTAX

Add-ASFilterCondition [[-Core] <ConnectCore>] [-FilterPolicyId] <int[]> [-FilterDefinition] <FilterDefinitionEnum> [-Operator] <FilterOperatorEnum> [-Value] <DateTime/long/bool/string> [-Unit] <FilterDefinitionItemSizeUnitEnum> [<CommonParameters>]

DESCRIPTION

The Add-ASFilterCondition cmdlet adds new filter condition for existing filter policy. You have to add value in according to filter definition type. These are DateTime (ArchivedDate,ItemDate), string (Path,RetentionCategory), long (ItemSize) and boolean (HasLegalHold). You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Common parameters:

FilterPolicyId, ConditionDefinition

Patameters according to ConditionDefinition value:

For ArchivedDate and ItemDate: Operator (Equal/YoungerThan/OlderThan/NotEqual/Like/NotLike), Value

For Path: Operator (Equal/NotEqual/Like/NotLike), Value

For RetentionCategory: Operator (Equal/GreaterEqual/GreaterThan/LessEqual/LesserThan/NotEqual/Like), Value

For HasShortcut : no more parameters

For ItemSize: Operator (Equal/NotEqual/Like/NotLike), Value, Unit (B, KiB, MiB, GiB)

For HasLegalHold: Value (True/False)

Example:

Add-ASFilterCondition -FilterPolicyId 3 -FilterDefinition HasShortcut	/* Creates new Filter Condition for specified Filter Policy */
$filtePolicy = Add-ASFilterPolicy -Name "ARCHIVED DATE 01/01/2016" Add-ASFilterCondition -FilterPolicyId $filtePolicy.FilterPolicyId -FilterDefinition ArchivedDate -Operator YoungerThan -Value "2016-1-1" Add-ASFilterCondition -FilterPolicyId $filtePolicy.FilterPolicyId -FilterDefinition ArchivedDate -Operator OlderThan -Value "2016-1-2"	/* First creates new Filter Policy then two new Filter Conditions to filter out items archived at 01/01/2016 only */
Get-ASFilterPolicy -Name "FPNAME" | Add-ASFilterCondition -FilterDefinition ItemDate -Operator Equal -Value "2016-1-1"	/* Creates new Filter Condition for specified Filter Policy */

Get-ASFileNamePolicy

SYNOPSIS

Get file name policy.

SYNTAX

Get-ASFileNamePolicy [-Core] <ConnectCore> -FileNamePolicyId] [<int> [[-Name] <string>] [<CommonParameters>]

DESCRIPTION

The Get-ASFileNamePolicy cmdlet gets file name policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASFileNamePolicy	/* Retrieves all file name policies */
Get-ASFileNamePolicy -Name Ownerless	/* Retrieves only specified file name policy */

Set-ASFileNamePolicy

SYNOPSIS

Set file name policy.

SYNTAX

Set-ASFileNamePolicy [-Core] <ConnectCore> -FileNamePolicyId] [<int>-Name] <string>] [[-NamingPolicy] [<string> [<CommonParameters>]

DESCRIPTION

The Set-ASFileNamePolicy cmdlet gets the file name policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASFileNamePolicy -FileNamePolicyId 1 -NamingPolicy "*archivename*_pstId-*pstid*_*pstnumber*.pst"	/* Sets naming policy of specified file name policy */

Add-ASFileNamePolicy

SYNOPSIS

Add new file name policy.

SYNTAX

Add-ASFileNamePolicy [-Core] <ConnectCore> [-Name] <string> -NamingPolicy] [<string> [<CommonParameters>]

DESCRIPTION

The Add-ASFileNamePolicy cmdlet creates a new file name policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASFileNamePolicy -Name OwnerlessCustom -NamingPolicy "*archivename*_pstId-*pstid*_*pstnumber*.pst"	/* Creates new file name policy */

Get-ASTargetPathNamePolicy

SYNOPSIS

Get Target Path Name Policy.

SYNTAX

Get-ASTargetPathNamePolicy [-Core] <ConnectCore> -TargetPathNamePolicyId][<int> [[-Name] <string>] [<CommonParameters>]

DESCRIPTION

The Get-ASTargetPathNamePolicy cmdlet gets the Target Path Name Policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASTargetPathNamePolicy Get-ASTargetPathNamePolicy -TargetPathNamePolicyId 42	/*Retrieves all Target Path Name Policies */ /*Retrieves specific Target Path Name Policy with id 42 */

Set-ASTargetPathNamePolicy

SYNOPSIS

Set Target Path Name Policy.

SYNTAX

Set-ASTargetPathNamePolicy [-Core] <ConnectCore> [-TargetPathNamePolicyId] <int> -Name] <string>] [[-NamingPolicy] [<string> [<CommonParameters>]

DESCRIPTION

The Set-ASTargetPathNamePolicy cmdlet sets the Target Path Name Policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. call Connect-ASCore once and core parameter will be automatically added from session state.

Example:

Set-ASTargetPathNamePolicy -TargetPathNamePolicyId 2 -Name testPS	/*Sets existing Target Path Name policy based on id */
Set-ASTargetPathNamePolicy -TargetPathNamePolicyId 2 -NamingPolicy *archivename*_*archiveid*	/*Sets existing Target Path Name policy tokes based on id */

Add-ASTargetPathNamePolicy

SYNOPSIS

Add Target Path Name Policy.

SYNTAX

Add-ASTargetPathNamePolicy [-Core] <ConnectCore> [-Name] <string> [-NamingPolicy] <string> [<CommonParameters>]

DESCRIPTION

The Add-ASTargetPathNamePolicy cmdlet creates a new Target Path Name Policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Examples:

Add-ASTargetPathNamePolicy -Name ARName&ID-NamingPolicy *archivename*_*archiveid*	/*Add new policy with specific name and tokes*/

Add-ASO365LeaversConfiguration

SYNOPSIS

PowerShell command for adding of Office 365 leavers configuration.

SYNTAX

Add-ASO365LeaversConfiguration [-Core] <ConnectCore> [-LinkId] <Guid?> [-LinkName] <string> [-SettingType] <Office365MailboxProvisioningSettingType> [-NamingScheme] <string> [-UsageLocation] <string> [-EMailSuffix]<string> [[-HideFromAddresslist] <SwitchParameter>] [[-BlockCredential] <SwitchParameter>] -LegalHoldType <Office365LegalHoldType> [-Office365LicenseId <int>] [-MaxRollingLicenses <int?>] [-DLNamingScheme <string>] -GroupMailboxNamingScheme <string> [-DeleteMailboxPermanently <bool?>] [<CommonParameters>]

DESCRIPTION

The Add-ASO365LeaversConfiguration creates leavers setting. Link names or linkIds can be used. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

$o365link = Get-ASLink -Type Office365 -LinkName "O365"	/* Creates Standard Office 365 Leavers configuration */

$o365license = Get-ASO365License -AccountSkuId "quadrotechlab3:ENTERPRISEPACK" Add-ASO365LeaversConfiguration -LinkId $o365link.LinkId -SettingType Leavers -NamingScheme "LEAVER_MapId_*containermappingID*_*archivename*" -UsageLocation UnitedStates -EMailSuffix "@domain.com" -HideFromAddresslist -BlockCredential -Office365LicenseId $o365license.Office365LicenseId -MaxRollingLicenses 5	/* Post Archive Shuttle 9.5, it is possible to create leavers configuration only with Litigation hold, which is set automatically by default */
Add-ASO365LeaversConfiguration -LinkId $o365link.LinkId -SettingType VirtualJournal -NamingScheme "VJOURNAL_MapId_*containermappingID*_*archivename*" -UsageLocation UnitedStates -EMailSuffix "@domain.com" -HideFromAddresslist -BlockCredential -Office365LicenseId $o365license.Office365LicenseId -MaxRollingLicenses 5	/* Creates Virtual Journal Leavers configuration */
Add-ASO365LeaversConfiguration -LinkId $o365link.LinkId -SettingType VirtualJournal -NamingScheme "MAORVJOURNAL_MapId_*containermappingID*_*archivename*" -UsageLocation UnitedStates -EMailSuffix "@domain.com" -HideFromAddresslist -BlockCredential -Office365LicenseId $o365license.Office365LicenseId -MaxRollingLicenses 5	/* Creates Journal Explosion Leavers configuration */

Get-ASO365License

SYNOPSIS

Get Archive Shuttle Office 365 license.

SYNTAX

Get-ASO365License [-Core] <ConnectCore> -Office365LicenseId] [<int[]>-AccountSkuId] [<string[]> [<CommonParameters>]

DESCRIPTION

The Get-ASO365License cmdlet gets Office 365 licenses. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASO365License	/* Retrieves all Office 365 licenses present within Archive Shuttle Database */
Get-ASO365License -AccountSkuId "quadrotechlab3:ENTERPRISEPACK" Get-ASO365License -Office365LicenseId 1	/* Retrieves only license with specified AccountSkuId */ /* Retrieves only license with specified ID (internal Archive Shuttle DB Id) */

Set-ASO365LeaversConfiguration

SYNOPSIS

PowerShell command for setting of Office 365 leavers configuration.

SYNTAX

Set-ASO365LeaversConfiguration [-Core] <ConnectCore> [-LinkId] <Guid?> [-LinkName] <string> [-SettingType] <Office365MailboxProvisioningSettingType> [[-NamingScheme] <string>] [[-UsageLocation] <Location?>] [[-EMailSuffix] <string>] [[-HideFromAddresslist] <bool?>] [[-BlockCredential] <bool?>] [-LegalHoldType <Office365LegalHoldType>] [-Office365LicenseId <int?>] [-MaxRollingLicenses <int?>] [-DLNamingScheme <string>] [-GroupMailboxNamingScheme <string>] [-DeleteMailboxPermanently <bool?>] [<CommonParameters>]

DESCRIPTION

The Set-ASO365LeaversConfiguration sets leavers setting. Link names or linkIds can be used. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASO365LeaversConfiguration -LinkName "O365" -SettingType Leavers -UsageLocation Slovakia	/* Sets UsageLocation of existing leavers configuration */
Set-ASO365LeaversConfiguration -LinkName "O365" -SettingType VirtualJournal -Office365LicenseId	/* Sets Rolling license count of existing leavers configuration */

$o365license.Office365LicenseId -MaxRollingLicenses 1	
Set-ASO365LeaversConfiguration -LinkName "O365" -SettingType VirtualJournal -DeleteMailboxPermanently 1	/* Sets Delete mailbox permanently */

Get-ASO365LeaversConfiguration

SYNOPSIS

Get Archive Shuttle Office 365 leavers configuration.

SYNTAX

Get-ASO365LeaversConfiguration [-Core] <ConnectCore> [[-LinkId] <Guid>] [[-LinkName] <string>] [[-SettingType] <Office365MailboxProvisioningSettingType>] [<CommonParameters>]

DESCRIPTION

The Get-ASO365LeaversConfiguration cmdlet gets the Office 365 leavers configuration. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASO365LeaversConfiguration	/* Retrieves all available Leavers configuration settings for all O365 links */
Get-ASO365LeaversConfiguration -LinkName "O365"	/* Gets leavers configuration of specific O365 link */
Get-ASO365LeaversConfiguration -SettingType Leavers	/* Gets all "Standard Leavers" configurations */

Get-ASModule

SYNOPSIS

Gets list of available modules.

SYNTAX

Get-ASModule [-Core] <ConnectCore> [[-ModuleId] <Guid>] [[-ModuleType] <ModuleTypeEnum>] [[-ComputerName] <string>] [[-Is64bit] <bool>] [[-CustomSchedule] <bool>] [[-IsActive] <bool>] [[-TimeToNextSchedule] <bool>] [<CommonParameters>]

DESCRIPTION

The Get-ASModule cmdlet gets a list of available Modules. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASModule -ModuleId "47d0ccbe-7638-e611-813a-005056b84ed8"	
Get-ASModule -ModuleType EVExport| Format-List ModuleId,ComputerName,ServiceName	/* Retrieves all Enterprise Vault Export modules with filtered list of three columns */
Get-ASModule-ModuleTypeADCollector| Where-Object {$_.DomainName -eq "domain1"}	/* Retrieves all Active Directory Collector modules for domain1 */ (new result property added in 7.7)
Get-ASModule -CustomSchedule 1 -IsActive 1	/* Retrieves only modules with custom schedule which are currently active */
Get-ASModule | Where-Object {($_.isStopped -eq 'True') -and ($_.LastStoppedDateUTC -like '*2020*')}| Start-ASModule	/*Retrieves only stopped modules from specific year and start these modules */

Set-ASModule

SYNOPSIS

Set module.

SYNTAX

Set-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [[-IsPerformanceStatisticsEnabled] <bool>] [[-DoUpdateModule] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Set-ASModule cmdlet set module. After the command is executed, the module version can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASModule -ModuleId ffed0c15-69e9-e811-8392-005056982247 -DoUpdateModule Set-ASModule -ModuleId	/* Sets column DoUpdate = TRUE in dbo.Module table */ /* Sets column DoUpdate = TRUE in dbo.Module table for array of modules */

ffed0c15-69e9-e811-8392-005056982247,aaed0c15-69e9-e811-8392-005056982247 -DoUpdateModule		
Set-ASModule -ModuleId ffed0c15-69e9-e811-8392-005056982247 -IsPerformanceStatisticsEnabled 1	/* Enables performance statistics collection for specified module */	
Get-ASModule | Set-ASModule -DoUpdateModule Get-ASModule | Set-ASModule -IsPerformanceStatisticsEnabled 1	/* Sets column DoUpdate = TRUE in dbo.Module table for all retrieved modules */ /* Enables performance statistics collection for all retrieved modules */	

Set-ASModuleLogLevel

SYNOPSIS

Sets module log level.

SYNTAX

Set-ASModuleLogLevel [-Core] <ConnectCore> [-ModuleId] <Guid[]> [-CoreLogLevel] <LogLevelEnum> [-ModuleLogLevel] <LogLevelEnum> [<CommonParameters>]

DESCRIPTION

The Set-ASModuleLogLevel cmdlet sets module log level. After the command is executed, the module log level can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASModuleLogLevel -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9" -CoreLogLevel Trace -ModuleLogLevel Trace	/* Sets TRACE level for specified module logs */
Get-ASModule -ModuleType Office365Import | Set-ASModuleLogLevel -CoreLogLevel Trace -ModuleLogLevel Trace	/* Sets TRACE level for all Office365 modules logs */

Get-ASModule | Set-ASModuleLogLevel -CoreLogLevel Trace -ModuleLogLevel Trace	/* Sets TRACE level for all existing modules logs */

Get-ASModulePerformance

SYNOPSIS

Gets list of module performance records.

SYNTAX

Get-ASModulePerformance [-Core] <ConnectCore> [[-ModuleId] <Guid[]>] [[-FromTimeStamp] <DateTime>] [<CommonParameters>]

DESCRIPTION

The Get-ASModulePerformance cmdlet gets a list of modules statistics. If FromTimeStamp parameter is not specified not set it will return latest record of module performance. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASModulePerformance -ModuleId a5eb1281-de91-e611-8147-005056b84ed8 -FromTimeStamp "2016-10-24 09:00:00"	/* Retrieves module performance records of specified module (time range: from specified UTC datetime to the latest record) */
Get-ASModulePerformance -ModuleId a5eb1281-de91-e611-8147-005056b84ed8	/* Retrieves latest module performance record of specified module */
Get-ASModule -ModuleType Office365Import | Get-ASModulePerformance | Select-Object -Property ModuleId, ServerCPU*	/* Retrieves latest performance records of all O365 modules with filtered view of specified properties only */

NOTE: Since Archive Shuttle 7.10, the "-LatestRecord" parameter is not supported anymore. It is automatically applied unless "-FromTimeStamp" parameter is defined.

Enable-ASModule

SYNOPSIS

Enable module.

SYNTAX

Enable-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Enable-ASModule cmdlet enables module. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are

mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Enable-ASModule -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9"	
Get-ASModule -ModuleType DellArchiveManager | Enable-ASModule	

Disable-ASModule

SYNOPSIS

Disable module.

SYNTAX

Disable-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Disable-ASModule cmdlet disables module. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Disable-ASModule -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9"	
Get-ASModule -ModuleType DellArchiveManager | Disable-ASModule	

Start-ASModule

SYNOPSIS

Start module.

SYNTAX

Start-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Start-ASModule cmdlet start module. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASModule -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9"	
Get-ASModule -ModuleType DellArchiveManager | Start-ASModule	/* Retrieves specified module and starts it (it can have small delay) */

Stop-ASModule

SYNOPSIS

Stop module.

SYNTAX

Stop-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Stop-ASModule cmdlet stop module. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Stop-ASModule -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9"	
Get-ASModule -ModuleType DellArchiveManager | Stop-ASModule	

Restart-ASModule

SYNOPSIS

Restart module.

SYNTAX

Restart-ASModule [-Core] <ConnectCore> [-ModuleId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Restart-ASModule cmdlet restart module. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Restart-ASModule -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9"	
Get-ASModule -ModuleType DellArchiveManager | Restart-ASModule	/* Retrieves specified module and restarts it (it can have small delay) */

Start-ASCommand

SYNOPSIS

Start command immediately.

SYNTAX

Start-ASCommand [-Core] <ConnectCore> [-Command] <CommandEnum[]> [<CommonParameters>]

DESCRIPTION

The Start-ASCommand cmdlet starts command immediately. After the command is executed, the result can be checked in the Archive Shuttle user interface/logs. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASCommand -Command Office365CollectLicenses	/* Command to collect Office 365 licenses is set to be executed */

Get-ASCommandQueueStatus

SYNOPSIS

Returns status of command.

SYNTAX

Get-ASCommandQueueStatus -Core <ConnectCore> [[-CommandId] <int>] [[-ModuleId] <Guid[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASCommandQueueStatus returns command queue status. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASCommandQueueStatus	/* Retrieves statuses of all commands within CommandQueue table */
Get-ASCommandQueueStatus -CommandId 801	/* Retrieves status of specified command */
Get-ASCommandQueueStatus -ModuleId b305cf97-f93d-e611-813c-005056b84ed8 -CommandId 801	/* Retrieves status of specified command related to specified module */
Get-ASCommandQueueStatus | Where-Object {$_.TaskExecutionStatusId -eq "Success"}	/* Retrieves only successfully processed commands */

Set-ASCommandIntervalLastExecutedDate

SYNOPSIS

Command for seting last executed date UTC of Command interval execution status table.

SYNTAX

Set-ASCommandIntervalLastExecutedDate -Core <ConnectCore> [-CommandId] [<int[]>] -Date] [<DateTime> [-All] [<SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Set-ASCommandIntervalLastExecutedDate cmdlet sets last executed date UTC of Command interval execution status table. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASCommandIntervalLastExecutedDate -CommandId 801 -Date "2222-02-02 12:00:00.000"	/*ADAM will set future date to prevent commands to be executed once default command interval is to be applied */

Get-ASScheduledTask

SYNOPSIS

Get info about scheduled task.

SYNTAX

Get-ASScheduledTask [-Core] <ConnectCore> [[-ScheduledTask] <ScheduledTaskEnum[]>] [[-ScheduledTaskId] <int[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASScheduledTask will return info about scheduled task. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASScheduledTask	/* Returns all Scheduled Tasks */
Get-ASScheduledTask -ScheduledTask AutoRestartModule	/* Returns Scheduled Task base on name */
Get-ASScheduledTask -ScheduledTaskId 33	/* Returns Scheduled Task base on TaskID */
Get-ASScheduledTask -ScheduledTaskId 1 -ScheduledTask AutoRestartModule	/* Composite filter base on TaskID and Task Name */

Set-ASScheduledTask

SYNOPSIS

Sets a scheduled task.

SYNTAX

Set-ASScheduledTask [-Core] <ConnectCore> -ScheduledTask] <ScheduledTaskEnum[]>] [[-ScheduledTaskId] <int[]>] [[-IsEnabled] [<bool?> -RunNow] <SwitchParameter>] [[-RepeatEvery] [<int?> [<CommonParameters>]

DESCRIPTION

The Set-ASScheduledTask will set a scheduled task. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASScheduledTask -ScheduledTask AutoEnableJournalTransformationUsers -RepeatEvery 10 -IsEnabled $true -RunNow	/* Sets particular task based on name*/
Set-ASScheduledTask -ScheduledTaskid 33 -RepeatEvery 10 -IsEnabled $true -RunNow	/* Sets particular task based on TaskId */
Set-ASScheduledTask -ScheduledTaskid 33 -ScheduledTask AutoEnableJournalTransformationUsers -RepeatEvery 10 -IsEnabled $true -RunNow	/* Sets 2 scheduled tasks id 33 and autoenableJTusers */
$scheduledTasks = Get-ASScheduledTask | Select-Object -ExpandProperty ScheduledTask	/* Sets all scheduled Tasks with values */

Set-ASScheduledTask -ScheduledTask AutoEnableJournalTransformationUsers -RepeatEvery 10 -IsEnabled $true -RunNow	/* Sets particular task based on name*/
$scheduleTasks | ForEach-Object{Set-ASScheduledTask -ScheduledTask $scheduleTasks -IsEnabled $true -RunNow}	

Enable-ASActiveDirectorySync

SYNOPSIS

Enables Active Directory sync.

SYNTAX

Enable-ASActiveDirectorySync [-Core] <ConnectCore> [-DistinguishedName] <string[]> [<CommonParameters>]

DESCRIPTION

The Enable-ASActiveDirectorySync cmdlet enables Active Directory sync. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Enable-ASActiveDirectorySync -DistinguishedName "DC=qa,DC=lab,DC=quadrotech-it,DC=com"

Disable-ASActiveDirectorySync

SYNOPSIS

Disable Active Directory sync.

SYNTAX

Disable-ASActiveDirectorySync [-Core] <ConnectCore> [-DistinguishedName] <string[]> [<CommonParameters>]

DESCRIPTION

The Disable-ASActiveDirectorySync cmdlet disables Active Directory sync. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Disable-ASActiveDirectorySync -DistinguishedName "DC=qa,DC=lab,DC=quadrotech-it,DC=com"

Get-ASADGroup

SYNOPSIS

Gets Active Directory groups.

SYNTAX

Get-ASADGroup -Core <ConnectCore> -DomainADContainerId] <int[]>] [[-GroupSid] <SecurityIdentifier[]>] [[-SAMAccountName] <string[]>] [[-IsEnabled] <bool]>] [[-All] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Get-ASADGroup returns Active Directory groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASADGroup	/* Returns first 1000 groups (due to default paging value=1000) */
Get-ASADGroup -All -IncludeTotalCount	/* Returns all Active Directory groups within Archive Shuttle database and shows total count of results */
Get-ASADGroup -DomainADContainerId 1 -GroupSid S-1-5-21-141886343-4149337270-1996687478-9161	
Get-ASADGroup -DomainADContainerId 1 -IsEnabled 1 -First 10 -Skip 10	/* Returns 10 groups (skips first 10) within specified domain which are active (IsEnabled=0 in AS DB = Group was deleted in AD) */

Get-ASADGroupMember

SYNOPSIS

Gets Active Directory group members.

SYNTAX

Get-ASADGroupMember -Core <ConnectCore> -GroupSid] <SecurityIdentifier[]>] [[-MemberSid] <SecurityIdentifier[]>] [[-MemberType] <GroupMemberType>] [[-All] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Get-ASADGroupMember returns Active Directory group members. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASADGroupMember	/* Returns first 1000 group members (due to default paging value=1000) */
Get-ASADGroupMember -All -IncludeTotalCount	/* Returns all group members with total count information */
Get-ASADGroupMember -MemberType Group	
Get-ASADGroupMember -GroupSid S-1-5-21-141886343-4149337270-1996687478-512	/* Returns all members of the specified group */
Get-ASADGroup -SAMAccountName "Domain Admins" -DomainADContainerId 4 | Get-ASADGroupMember -MemberType User -IncludeTotalCount	/* Returns all user members of specified group within specified domain and total count information */

Get-ASADDomain

SYNOPSIS

Get all domains.

SYNTAX

Get-ASADDomain [-Core] <ConnectCore> [<CommonParameters>]

DESCRIPTION

The Get-ASADDomain returns all domains. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Start-ASSyncADDomains

SYNOPSIS

Start AD domains synchronization.

SYNTAX

Start-ASSyncADDomains -Core <ConnectCore> [<CommonParameters>]

DESCRIPTION

The Start-ASSyncADDomains starts synchronization of Active Directory domains. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Get-ASExchangeServer

SYNOPSIS

Gets Exchange Servers.

SYNTAX

Get-ASExchangeServer -Core <ConnectCore> [-ADContainerId [<int[]>]] [<CommonParameters>]

DESCRIPTION

The Get-ASExchangeServer returns Exchange servers. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASExchangeServer	
Get-ASExchangeServer -ADContainerId 2	/* As of Archive Shuttle 7.7, it is possible to define only the ADContainerId parameter. This is a custom command for ADAM, and in case other parameters are required, it should be logged in TFS */
Get-ASExchangeServer|Format-Table ExchangeServerId,AdContainerId,Name,FullyQualifiedDomainName -AutoSize	
Get-ASExchangeServer | Where-Object {$_.ExchangeServerId -eq "1"}	

Get-ASExchangeDatabase

SYNOPSIS

Gets Exchange databases.

SYNTAX

Get-ASExchangeDatabase -Core <ConnectCore> [-ExchangeServerId [<int[]>]] [<CommonParameters>]

DESCRIPTION

The Get-ASExchangeDatabase return Exchamge databases. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASExchangeDatabase -ExchangeServerId 1	/* As of Archive Shuttle 7.7, it is possible to define only ExchangeServerId parameter. This is a custom command for ADAM, and in case other parameters are required, it should be logged in TFS */
Get-ASExchangeServer | Where-Object {$_.ExchangeServerId -eq "1"} | Get-ASExchangeDatabase | Format-Table ExchangeServerId,Name,DistinguishedName -AutoSize	

Add-ASEnvironment

SYNOPSIS

Creates a new environment.

SYNTAX

Add-ASEnvironment [-Core] <ConnectCore> [-EnvironmentType] <ContainerTypeEnvironmentEnum> [-ModuleId] <Guid> [-DisplayName] <string> [-SqlServer] <string> [-SqlDatabaseName] <string> [[-SqlTableSchema] <string>] [[-LinkSqlServer] <string>] [[-BasePaths] <string>]

[-WebServerUrl] <string> [<CommonParameters>]

DESCRIPTION

The Add-ASEnvironment cmdlet creates a new environment. After the command is executed, the environment can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and core parameter will be automatically added from session state.

Parameters (M)EnvironmentType, (V)ModuleId, (V)DisplayName, (V)SqlServer, (V) SqlDatabaseName are common for all containers. Other parameters depends on environment type:

- For type DellArchiveManager: (V)LinkSqlServer.
- For type SourceOne: (V)LinkSqlServer, (V)BasePaths.
- For type EasDynamicParameters: (V)SqlTableSchema, (V)LinkSqlServer, (V)WebServerUrl.

NOTE: Some parameters are dynamically added/removed (based on EnvironmentType)

Example:

Add-ASEnvironment -EnvironmentType EnterpriseVault -DisplayName "Enterprise Vault 12" -ModuleId "434b9603-1ab9-e511-80e6-005056b82cc9" -SqlServer "SQLSERVER01\INSTANCE1" -SqlDatabaseName "EnterpriseVaultDirectory"	
$evcolmodule = Get-ASModule -ModuleType EVCollector \| Where-Object {$_.ComputerName -eq "Computer01"} Add-ASEnvironment -EnvironmentType EnterpriseVault -DisplayName "Enterprise Vault 12" -ModuleId $evcolmodule.ModuleId -SqlServer "SQLSERVER01\INSTANCE1" -SqlDatabaseName "EnterpriseVaultDirectory"	

Set-ASEnvironment

SYNOPSIS

Sets an existing environment.

SYNTAX

Set-ASEnvironment [-Core] <ConnectCore> [-EnvironmentType] <ContainerTypeEnvironmentEnum> [-ModuleId] <Guid> [-DisplayName] <string> [-SqlServer] <string> [-SqlDatabaseName] <string> [[-SqlTableSchema] <string>] [[-LinkSqlServer] <string>] [[-BasePaths] <string>] [-EnvironmentId] <int> [<CommonParameters>]

DESCRIPTION

The Set-ASEnvironment cmdlet sets an existing environment. After the command is executed, the environment can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Parameters (M)EnvironmentId, (M)EnvironmentType, (V)ModuleId, (V)DisplayName, (V) SqlServer, (V)SqlDatabaseName are common for all containers. Other parameters depends on environment type:

- For type DellArchiveManager: (V)LinkSqlServer.
- For type SourceOne: (V)LinkSqlServer, (V)BasePaths.
- For type EasDynamicParameters: (V)SqlTableSchema, (V)LinkSqlServer, (V)WebServerUrl.

NOTE: Some parameters are dynamically added/removed (based on EnvironmentType)

Example:

Set-ASEnvironment -EnvironmentId 1 -EnvironmentType EnterpriseVault -ModuleId "C0484EE5-CB39-E611-813B-005056B84ED8" -DisplayName "EV10 QA" -SqlServer "172.16.7.8\INSTANCE1" -SqlDatabaseName "EnterpriseVaultDirectory"	

Get-ASEnvironment

SYNOPSIS

Displays data regarding the environment.

SYNTAX

Get-ASEnvironment [[-Core] <ConnectCore>] [[-EnvironmentType] <ContainerTypeEnvironmentEnum[]>] [[-DisplayName] <string[]>] [[-SqlServerName] <string[]>] [[-SqlDatabaseName] <string[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASEnvironment cmdlet gets data regarding a specified environment. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASEnvironment	/* Returns all existing environments */
Get-ASEnvironment -EnvironmentType EnterpriseVault	/* Returns only Enterprise Vault environments*/
Get-ASEnvironment -DisplayName "EV10 QA" -SqlServerName "172.16.7.8\INSTANCE1"	/* Returns only specific Enterprise Vault environment */

Get-ASVaultStore

SYNOPSIS

Gets list of Vault stores.

SYNTAX

Get-ASVaultStore [-Core] <ConnectCore> [[-LinkId] <Guid[]>] [[-VaultStoreId] <string[]>] [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASVaultStore cmdlet gets a list of available Vault Stores. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink -LinkName "VS1" | Get-ASVaultStore	/* Enterprise Vault Vault Store received through link name */
Get-ASLink -Type EnterpriseVault | Where-Object {$_.LinkName -like "*VS1*"} | Get-ASVaultStore	/* Enterprise Vault Vault Store received through link type and specific name */

Enable-ASArchiveGathering

SYNOPSIS

Enables archive gathering.

SYNTAX

Enable-ASArchiveGathering [-Core] <ConnectCore> [[-LinkId] <Guid[]>] [[-VaultStoreId] <string[]>] [<CommonParameters>]

DESCRIPTION

The Enable-ASArchiveGathering cmdlet enables archive gathering. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink -LinkName "VS1" | Get-ASVaultStore | Enable-ASArchiveGathering	/* Enables archive gathering for specified Enterprise Vault Vault Store through link name */
Enable-ASArchiveGathering -LinkId "434b9603-1ab9-e511-80e6-005056b82cc9" Enable-ASArchiveGathering -VaulStoreId "13851C140C2F8184F993AF0C26BDA3C6C1210000ev10-site"	
$evlink = Get-ASLink -Type EnterpriseVault \| Where-Object {$_.LinkName -like "*VS1*"} | Get-ASVaultStore Enable-ASArchiveGathering -VaultStoreId $evlink.VaultStoreId	/* First Enterprise Vault Vault Store is searched through Link name then archive gathering enabled

NOTE: Currently, only Enterprise Vault is supported (7.7).

Disable-ASArchiveGathering

SYNOPSIS

Disables archive gathering.

SYNTAX

Disable-ASArchiveGathering [-Core] <ConnectCore> [[-LinkId] <GUID>] [[-VaultStoreId] <string>] [<CommonParameters>]

DESCRIPTION

The Disable-ASArchiveGathering cmdlet disables archive gathering. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink -LinkName "VS1" | Get-ASVaultStore | Disable-ASArchiveGathering	/* Disables archive gathering for specified Enterprise Vault Vault Store through link name */
Disable-ASArchiveGathering -LinkId "434b9603-1ab9-e511-80e6-005056b82cc9" Disable-ASArchiveGathering -VaulStoreId "13851C140C2F8184F993AF0C26BDA3C6C1210000ev10-site"	
$evlink = Get-ASLink -Type EnterpriseVault \| Where-Object {$_.LinkName -like "*VS1*"}| Get-ASVaultStore Disable-ASArchiveGathering -VaultStoreId $evlink.VaultStoreId	/* First Enterprise Vault Vault Store is searched through Link name then archive gathering disabled

NOTE: Currently, only Enterprise Vault is supported (7.7).

Start-ASArchiveGathering

SYNOPSIS

Start archive gathering.

SYNTAX

Start-ASArchiveGathering [-Core] <ConnectCore> [[-LinkId] <Guid[]>] [[-VaultStoreId] <string[]>] [<CommonParameters>]

DESCRIPTION

The Start-ASArchiveGathering cmdlet starts archive gathering. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink -LinkName "VS1" | Get-ASVaultStore | Start-ASArchiveGathering	/* Starts archive gathering for specified Enterprise Vault Vault Store through link name */
Start-ASArchiveGathering -LinkId "434b9603-1ab9-e511-80e6-005056b82cc9" Start-ASArchiveGathering -VaulStoreId "13851C140C2F8184F993AF0C26BDA3C6C1210000ev10-site"	
$evlink = Get-ASLink -Type EnterpriseVault \| Where-Object {$_.LinkName -like "*VS1*"}| Get-ASVaultStore Start-ASArchiveGathering -VaultStoreId $evlink.VaultStoreId	/* First Enterprise Vault Vault Store is searched through Link name, then archive gathering starts.

NOTE: Currently, only Enterprise Vault and EAS is supported (7.7).

Get-ASLicense

SYNOPSIS

Get information about licenses.

SYNTAX

Get-ASLicense [-Core] <ConnectCore> [<CommonParameters>]

DESCRIPTION

The Get-ASLicense will return info about licenses. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLicense	/* Returns all information of Archive Shuttle license(s)

Add-ASLink

SYNOPSIS

Creates a new link.

SYNTAX

Add-ASLink [-Core] <ConnectCore> [-EnvironmentType] <ContainerTypeLinkEnum> [-LinkName] <string> [-LinkPath] <string> [-CustomerName] <string> [-NumberOfWorkers] <int> [-BatchSizeMb] <int> [-Format] <MailNativeFormatEnum> [-DefaultRolloverBytes] <long?>

[-DefaultRolloverItemCount <int?>] [-OutputPath] <string> [-LowWatermark <long?>] [-HighWatermark <long?>] [-FileNamePolicyId <int?>] -StorageType <StorageType> -Path <string> -AmazonS3Account <string> -AzureBlobStorage <string> [<CommonParameters>]

DESCRIPTION

The Add-ASLink cmdlet creates a new link. For link type=PST, PST Output path watermarks can be specified. After the command is executed the link can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Common Parameters: EnvironmentType, LinkName. Other parameters depends on environment type.

- For type ProofPoint: LinkPath, CustomerName, NumberOfWorkers, BatchSizeMb.
- For type PST or PSTJournalExplosion: DefaultRolloverBytes, DefaultRolloverItemCount, OutputPath, LowWatermark, HighWatermark.
- For type PSTJournalExplosion: FileNamePolicyId

NOTE: Some parameters are dynamically added/removed (based on EnvironmentType)

Example:

Add-ASLink -EnvironmentType Office365 -LinkName "O365"	/* Creates new Office 365 link */
Add-ASLink -EnvironmentType PST -Format PST -LinkName "PST as TARGET" -OutputPath \ \UNCOutputPath-LowWatermark 21474836480 -HighWatermark 21474836480	/* Creates PST Target link with PST Output path and its watermarks specified */
Add-ASLink -EnvironmentType Office365 -LinkName "O365"	/* Creates new Office 365 link */
Add-ASLink -EnvironmentType PST -Format PST -LinkName "PST as TARGET" -OutputPath \	/* Creates PST Target link with PST Output path and it's watermarks specified */

\UNCOutputPath-LowWatermark 21474836480 -HighWatermark 21474836480	
Add-ASLink -EnvironmentType StorageImport -LinkName "Storage-Amazon" -StorageType AmazonS3 -AmazonS3Account AmazonS3Account -Path "NewBucketUser01\Folder01"	/* Creates Amazon Storage Import link */

Get-ASLink

SYNOPSIS

Gets list of available links.

SYNTAX

Get-ASLink [-Core] <ConnectCore> [[-LinkId] <Guid>] [[-LinkName] <string>] [[-ComputerName] <string>] [[-Type] <ContainerTypeEnum>] [[-HasLinkDatabase] <bool>] [[-IsArchiveGatheringEnabled] <bool>] [<CommonParameters>]

DESCRIPTION

The Get-ASLink cmdlet gets a list of available Links. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink	
Get-ASLink -HasLinkDatabase 1 | ft LinkName	
Get-ASLink -Type EnterpriseVault -LinkName "LINK01"-ComputerName "COMPUTER01" -HasLinkDatabase 1 -IsArchiveGatheringEnabled 1 -IncludeTotalCount	
Get-ASLink -Type EnterpriseVault | Where-Object {$_.LinkName -like "*LINK01"}	

Get-ASLinkHealth

SYNOPSIS

Gets source link health details.

SYNTAX

Get-ASLinkHealth [-Core] <ConnectCore> [[-LinkId] <Guid[]>] [[-LinkName] <string[]>] [[-ContainerType] <ContainerTypeEnum[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASLinkHealth cmdlet returns details of source links health (e.g. staging area free space, percentage used, watermarks etc). You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLinkHealth|Select-Object LinkName,ContainerType,StagingArea,StagingFreeSpace,TotalDiskSpace,PercentUsed| Format-Table -AutoSize	/* Returns health details of all source links in formatted table with specific columns only*/
Get-ASLinkHealth -LinkId "68403d39-b89d-46f8-90e8-78d5e974e25d"	/* Returns full health details of specified source link*/

Get-ASLinkModule

SYNOPSIS

Gets modules assigned to link.

SYNTAX

Get-ASLinkModule [-Core] <ConnectCore> [-LinkId] <Guid[]> [<CommonParameters>]

DESCRIPTION

The Get-ASLinkModule cmdlet gets a list of modules assigned to link. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASLink -LinkId "62aa5b86-cd39-e611-813b-005056b84ed8"	
Get-ASLink | Get-ASLinkModule | Sort-Object ServiceName |Format-Table ServiceName,ModuleId,ComputerName -AutoSize	

Set-ASLinkModule

SYNOPSIS

Sets module to link.

SYNTAX

Set-ASLinkModule [-Core] <ConnectCore> [-LinkId] <Guid[]> [-ModuleId] <Guid> [<CommonParameters>]

DESCRIPTION

The Set-ASLinkModule cmdlet assigns module to link. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASLinkModule -LinkId "9cdb67de-17f3-4d79-9822-41bfdb4c9007" -ModuleId "f9c9466-cc39-e611-813b-005056b84ed8"	
Get-ASLink -LinkName "EV12" | Set-ASLinkModule -ModuleId "f9c9466-cc39-e611-813b-005056b84ed8"	
$evexmodule = Get-ASModule -ModuleType EVExport \| Where-Object {$_.ComputerName -eq "COMPUTER01"} Get-ASLink -Type EnterpriseVault | Where-Object {$_.LinkName -eq "EV12"} | Set-ASLinkModule -ModuleId $evexmodule.ModuleId	

Remove-ASLinkDatabase

SYNOPSIS

Remove Archive Shuttle link databases.

SYNTAX

Remove-ASDatabase [-Core] <ConnectCore> [-LinkId] <Guid[]> [-LinkName] <string[]> [-DeleteAll] <SwitchParameter> [[-DeleteHistoryDB] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Remove-ASLinkDatabase cmdlet remove AS link databases. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASLinkDatabase -LinkName "LinkName01"	/* Removes link database of specified link */
Remove-ASLinkDatabase -LinkName "LinkName01" -DeleteHistoryDB	/* Removes link databases of specified link (including history database if exists) */
Remove-ASLinkDatabase -DeleteAll	/* Removes all link databases at related Archive Shuttle Database Instance */

Add-ASItemDatabase

SYNOPSIS

Creates item database.

SYNTAX

Add-ASItemDatabase [-Core] <ConnectCore> -SqlServerName] [<string> [-LinkId] [<string>] [-LinkName] [<string>] [<CommonParameters>]

DESCRIPTION

The Add-ASItemDatabase cmdlet creates item database for link(s). Link names or linkIds can be used. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

$evlink = Get-ASLink -Type EnterpriseVault \| Where-Object {$_.LinkName -eq "*LinkName01*"} Add-ASItemDatabase -SqlServerName "172.0.0.0\INSTANCE1" -LinkId $evlink.LinkId	

Set-ASStagingAreaPath

SYNOPSIS

Sets staging area.

SYNTAX

Set-ASStagingAreaPath [-Core] <ConnectCore> -Path] <string>] [[-Default] [<SwitchParameter>-LinkId] <Guid[]>] [[-Azure] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Set-ASStagingAreaPath cmdlet sets staging area. After the command is executed, the result can be checked in the Archive Shuttle user interface. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASStagingAreaPath -LinkId "d2f8f070-263c-e611-813b-005056b84ed8" -Path "\\UNCPath" Set-ASStagingAreaPath -Path "\\UNCPath" -Default Get-ASLink -LinkName "LinkName1"| Set-ASStagingAreaPath -Path "\\UNCPath"	
$evlink = Get-ASLink -Type EnterpriseVault \| Where-Object {$_.LinkName -eq "LinkName1"} Set-ASStagingAreaPath -Path "\\UNCPath" -LinkId $evlink.LinkID -Default	

Start-ASStagingAreaCleanup

SYNOPSIS

Starts staging area cleanup for inserted LinkIds/ContainerMappingIds.

SYNTAX

Start-ASStagingAreaCleanup [-Core] <ConnectCore> [-ContainerMappingIds] <int[]> [-LinkIds] <Guid[]> [-ItemAge] <ItemAgeEnum> [-IncludeDeletedMappings] <SwitchParameter> -SkipValidation] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Start-ASStagingAreaCleanup cmdlet starts staging area cleanup based on inserted parameters. There is option to start cleanup based on ContainerMappingIds or LinkIds. Parameter ItemAge defines items with certain age which will be taken into account for deletion. This parameter is also related to System Configuration setting ClearStagingAreaFilesOlderThan [242]. For the option ContainerMappingIds, there is additional parameter SkipValidation; items from Staging area will be deleted without core check, so also items which were not ingested yet and have NULL value stored for columns ImportStartedDateUtc and ImportedDateUtc in ItemRouting table will be deleted from Staging area.

In case LinkIds are used for cleanup there is additional parameter IncludeDeletedMappings.

Example:

Start-ASStagingAreaCleanup -ContainerMappingId 1 -ItemAge OlderThanConfiguredHours Start-ASStagingAreaCleanup -ContainerMappingId 1,2,3 -ItemAge All Start-ASStagingAreaCleanup -ContainerMappingId 1,2,3 -ItemAge All -SkipValidation	/* Removes legacy files of certain age from staging area of specified container mapping */ /* Removes legacy files from staging area of specified list of container mappings */ /* Removes legacy files from staging area of specified list of container mappings without core check */
Start-ASStagingAreaCleanup -LinkId e438fb86-4574-490a-8bc4-4d3ec58c65be -ItemAge All Start-ASStagingAreaCleanup -LinkId e438fb86-4574-490a-8bc4-4d3ec58c65be -ItemAge All -IncludeDeletedMappingItems	/* Removes all legacy files from staging area of specified link. Basically it will find all container mappings related to specified link and cleans the legacy files */ /* Removes all legacy files from staging area of specified link. and includes also data of already deleted mappings */
Get-ASLink -Type EnterpriseVault | ForEach-Object {Start-ASStagingAreaCleanup -LinkId $_.LinkId -ItemAge OlderThanConfiguredHours -IncludeDeletedMappingItems}	/* Retrieves all EnterpriseVault links and then removes all legacy files from staging area (files of deleted mappings included) */

Get-ASProgressStatistics

SYNOPSIS

Get Progress and Performance Statistics.

SYNTAX

Get-ASProgressStatistics [-Core] <ConnectCore> [-StatisticsType] <StatisticsTypeEnum>-LinkId] [<Guid?> [<CommonParameters>]

DESCRIPTION

The Get-ASProgressStatistics will return progress and performance based on defined parameters. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASProgressStatistics -StatisticsType BasicStatistics Get-ASProgressStatistics -StatisticsType OverallPerformance	/* Retrieves all Basic Statistics */ /* Retrieves all Overall Performance Statistics */

Get-ASProgressStatistics -StatisticsType BasicStatistics -LinkId e438fb86-4574-490a-8bc4-4d3ec58c65be Get-ASProgressStatistics -StatisticsType OverallPerformance -LinkId e438fb86-4574-490a-8bc4-4d3ec58c65be	/* Retrieves Basic Statistics for particular link */ /* Retrieves Overall Performance Statistics for particular link */
Get-ASProgressStatistics -StatisticsType OverallPerformance -LinkId e438fb86-4574-490a-8bc4-4d3ec58c65be | Select-object -Property *Size*	/* Retrieves Overall Performance Statistics for particular link and returns only size statistics for it */

Add-ASPstSourcePath

SYNOPSIS

Add the PST source for link.

SYNTAX

Add-ASPstSourcePath [-Core] <ConnectCore> [-LinkId] <Guid> [-ModuleId] <Guid> [-Path] <string> [<CommonParameters>]

DESCRIPTION

The Add-ASPstSourcePath cmdlet adds PST source for PST link. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASPstSourcePath -LinkId "dab5f4a9-263c-e611-813b-005056b84ed8" -ModuleId "249170e0-253c-e611-813b-005056b84ed8" -Path "UNCPath"	
$link = Get-ASLink -Type PST \|Where-Object {$_.LinkName -eq "PSTSource"} $pstexportmodule = Get-ASModule -ModuleType PstExport \|Where-Object {$_.ComputerName -eq "ComputerName1"} Add-ASPstSourcePath -Path "\\UNCPath" -LinkId $link.LinkID -ModuleId $pstexportmodule.ModuleId	

Remove-ASPstSourcePath

SYNOPSIS

Remove the PST source for link.

SYNTAX

Remove-ASPstSourcePath [-Core] <ConnectCore> [-PstDirectoryId] <int[]> [<CommonParameters>]

DESCRIPTION

The Remove-ASPstSourcePath cmdlet deletes PST source for PST link. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASPstSourcePath -PstDirectoryId "1"	/* Removes record from [ArchiveShuttleDirectory].[dbo].[PstDirectory] table */

Set-ASPstSourcePath

SYNOPSIS

Set the PST source for link.

SYNTAX

Set-ASPstSourcePath [-Core] <ConnectCore> [-PstDirectoryId] <int[]> [-Path] <string> [<CommonParameters>]

DESCRIPTION

The Set-ASPstSourcePath cmdlet set PST source for PST link. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASPstSourcePath -PstDirectoryId "1" -Path "\\UNCPath"	/* Currently, it is possible to only change the path of existing PST directory record (LinkId or ModuleId not supported) */

Set-ASPstTargetPath

SYNOPSIS

Sets PST output path for target link.

SYNTAX

Set-ASPstTargetPath [-Core] <ConnectCore> [-LinkId] <Guid[]> [-Path] <string> [<CommonParameters>]

DESCRIPTION

The Set-ASPstTargetPath cmdlet sets PST output path of target link. After the command is executed, the result can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASPstTargetPath -LinkId "D9B5F4A9-263C-E611-813B-005056B84ED8" -Path "\\UNCPath"	
Get-ASLink -Type PST -LinkName "PST as TARGET QA" | Set-ASPstTargetPath -Path"\ \UNCPath"	

Set-ASPstTemporaryPath

SYNOPSIS

Sets PST temporary path for target link.

SYNTAX

Set-ASPstTemporaryPath [-Core] <ConnectCore> [-LinkId] <Guid[]>] [-Path] <string>] [-SetAsDefault] [<SwitchParameter>] [-ClearPath] [<SwitchParameter>[-ClearPath] [<SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Set-ASPstTemporaryPath cmdlet sets PST temporary path of target link. After the command is executed, the result can be checked in the Archive Shuttle user interface. Not all parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASPstTemporaryPath -LinkId "D9B5F4A9-263C-E611-813B-005056B84ED8"	/*Sets PST temporary path for specific linkId */

Get-ASLink -Type PST -LinkName "PST as TARGET QA" | Set-ASSPstTemporaryPath -Path"\\UNCPath"	/*Supports piping feature*/
Set-ASPstTemporaryPath -Path "\\UNCPath" -SetAsDefault	/*Set defined UNC Path as default for all PST type links*/
Set-ASPstTemporaryPath -LinkId "D9B5F4A9-263C-E611-813B-005056B84ED8" -ClearPath	/*Removes set UNC temporary path from PST link*/

Add-ASContainer

SYNOPSIS

Creates a new container.

SYNTAX

Add-ASContainer -Core] <ConnectCore>] [-ContainerType] <AddContainerTypeEnum> [-LinkId] <Guid> [-EWSId] <string> [-PublicFolderName] <string> [-FolderPath] <string> [-PrimarySMTPAddress] [<string>] [[-DistinguishedName] [<string>-GroupName] [<string>-UserSid] [<SecurityIdentifier> [-SharepointUrl] <string> [-SAMAccountName] <string> [-UserPrincipalName] <string> [-UserName] <string> [-TargetName] <string> [<CommonParameters>]

DESCRIPTION

The Add-ASContainer cmdlet creates a new container. Parameters (M)ContainerType and (M) LinkId are mandatory for all containers. Other parameters depends on container type.

- For types PublicFolder, ModernPublicFolder, ModernPublicFolderOnline: (M)EWSId, (M) PublicFolderName, (M)FolderPath, (V) PrimarySMTPAddress, (V) DistinguishedName, (V) TargetName.
- For types Office365GroupsMail: (M)GroupName, (M)PrimarySMTPAddress, (V) UserSid.
- For types Office365GroupsSharepoint: (M)GroupName, (M)PrimarySMTPAddress, (V)UserSid, (M)SharepointUrl.
- For types Office365, SharedMailbox, SharedMailboxOnline: (M)SAMAccountName, (V) DistinguishedName, (M)UserPrincipalName, (M)UserName, (M)PrimarySMTPAddress.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASContainer -ContainerType PublicFolder -EWSId "AAEuAAAAAAAaRHOQqmYRzZvIAKoAL8RaAwCes/4PRwIuRLqwkpDBpA9uAAAHoLkMAAA=" -	/* Creates public folder container in Archive Shuttle Directory database with specified parameters and links to specified Archive Shuttle link*/

PublicFolderName "PublicFolderName1" -FolderPath "\IPM_SUBTREE" -LinkId 963f8ef9-c6aa-42a0-8d32-45c5bc84d153 $pflink = Get-ASLink -Type PublicFolder \| Where-Object {$_.LinkName -like "*adam*"} Add-ASContainer -ContainerType PublicFolder -EWSId "AAEuAAAAAAAaRHOQqmYRzZvIAKoAL8RaAwCes/4PRwIuRLqwkpDBpA9uAAAHoLkMAAA=" -PublicFolderName "PublicFolderName1" -FolderPath "\IPM_SUBTREE" -LinkId $pflink.LinkId	/* Gets specified Archive Shuttle link first and then creates public folder container in Archive Shuttle Directory database with specified parameters and link to subjected Archive Shuttle link*/
$o365link = Get-ASLink -Type Office365 \| Where-Object {$_.LinkName -like "*O365*"} Add-ASContainer -ContainerType Office365GroupsMail -GroupName O365GROUP1 -LinkId $o365link.LinkId	/* Gets specified Office 365 Archive Shuttle link first and then creates Office365 Group container in Archive Shuttle Directory database with specified parameters and link to subjected Office 365 Archive Shuttle link*/
$o365link = Get-ASLink -Type Office365 \| Where-Object {$_.LinkName -like "*O365*"} Add-ASContainer -ContainerType Office365GroupsSharepoint -GroupName O365GROUP2 -SharepointUrl "https://sharepoint.com/sites/O365GROUP2" -LinkId $o365link.LinkId	/* Gets specified Office 365 Archive Shuttle link first and then creates Office365 Groups Sharepoint container in Archive Shuttle Directory database with specified parameters and link to subjected Office 365 Archive Shuttle link*/

Add-ASContainerToUser

SYNOPSIS

Adds container to user.

SYNTAX

Add-ASContainerToUser [-Core] <ConnectCore> [-ContainerId] <Guid> [-UserSID] <string>

[<CommonParameters>]

DESCRIPTION

The Add-ASContainerToUser will assign user to the container. You can specify core as a parameter, or call Connect-ASCore once and core parameter will be automatically added from session state.

Example:

Add-ASContainerToUser -ContainerId 'c26089cc-a81f-48ad-aa95-00783e2fe408' -UserSID 'S-1-5-21-2153452352-2279314742-1272584938-3645'	/* Assigns container with specified ContainerId to User with specified UserSID */

Get-ASArchive

SYNOPSIS

Get archive.

SYNTAX

Get-ASArchive [-Core] <ConnectCore> [-ContainerType] <ContainerTypeEnum> [[-ContainerId] <Guid[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASArchive will return existing archive(s) based on defined parameters. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASArchive -ContainerType EnterpriseVault	/* Retrieves all Enterprise Vault archives */
Get-ASArchive -ContainerType DellArchiveManager -ContainerId "ContainerId"	/* Retrieves information about specific archive */

Get-ASUser

SYNOPSIS

Gets Active Directory user.

SYNTAX

Get-ASUser -Core <ConnectCore> [[-UserSid] <SecurityIdentifier[]>] [[-SAMAccountName] <string[]>] [[-UserPrincipalName] <string[]>] [[-Email] <string[]>] [[-PrimarySMTPAddress] <string[]>] [[-DomainADContainerId] <int[]>] [[-All] <SwitchParameter>] [[-Skip] <uint64>] [[-First] <uint64>]

[<CommonParameters>]

DESCRIPTION

The Get-ASUser returns Active Directory user. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASUser	
Get-ASUser -All -IncludeTotalCount | Select-Object UserSid,SAMAccountName	
Get-ASUser -First 1 -Skip 1000	
Get-ASUser -DomainADContainerId 1 -SAMAccountName "SAName"	

Get-ASContainers

SYNOPSIS

Displays data regarding container(s).

SYNTAX

Get-ASContainers [[-Core] <ConnectCore>] [[-ContainerId] <Guid>] [[-LegacyExchangeDN] <string>] [[-ContainerType] <ContainerTypeEnum>] [[-Name] <String>] [-SAMAccountName] [<string[]>] [-TagName] <string[]> [-Skip <uint64>] [-First <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASContainers cmdlet gets data regarding a specified container or containers. The result contains: Container Id, Name, Link Id, Item Count, Item Size, and Enterprise Vault Environment Name. By default all containers details are displayed after the command executes. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASContainers -First 10| Select-Object Name,ItemCount|Sort-Object Name	
Get-ASContainers |Sort-Object Name|Select-Object Name -Last 5	
Get-ASContainers -All -IncludeTotalCount | Sort-Object Name|Select-Object -Property Name -First 6	
Get-ASContainers -IncludeTotalCount -All -Skip 1000	

Get-ASContainers -TagName TAG01	/* Gets list of containers assigned to specific Tag */

Get-AdamContainer

SYNOPSIS

Gets Adam container.

SYNTAX

Get-AdamContainer [[-Core] <ConnectCore>] [-ContainerType] <GetContainerTypeEnum> [-EWSId] <string[]> [-GroupName] <string[]> [-PrimarySMTPAddress] <string[]> [<CommonParameters>]

DESCRIPTION

The Get-ASContainer gets adam container. Parameter (M)ContainerType is mandatory for all containers. Other parameters depends on container type:

- For types PublicFolder, ModernPublicFolder, ModernPublicFolderOnline: (M)EWSId.
- For types Office365GroupsMail and Office365GroupsSharepoint: (M)GroupName.
- For types SharedMailbox, SharedMailboxOnline: (M)PrimarySMTPAddress.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-AdamContainer -ContainerType ModernPublicFolder -All -IncludeTotalCount	/* Retrieves all modern public folder containers and shows total count of these containers*/
Get-AdamContainer -ContainerType PublicFolder -EWSId	/* Retrieves specified public folder container based on EWSId*/
Get-AdamContainer -ContainerType Office365GroupsMail -GroupName GROUPNAME01	/* Retrieves specified Office 365 Group container based on group name*/
Get-AdamContainer -ContainerType SharedMailbox -PrimarySMTPAddress user1@domain1.onmicrosoft.com	/* Retrieves specified shared mailbox container based on email address*/

Get-ASMailbox

SYNOPSIS

Gets Archive Shuttle mailbox.

SYNTAX

Get-ASMailbox -Core <ConnectCore> [[-SAMAccountName] <string[]>] [[-UserSid] [[-UserPrincipalName] <string[]>]<SecurityIdentifier[]>] [[-MailboxType] <ExchangeMailboxTypeEnum>] [[-MailboxKind] <MailboxKindEnum>] [[-MailboxKind] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASMailbox returns mailboxes. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. All Parameters are accepted as composite filter.

Example:

Get-ASMailbox	
Get-ASMailbox -All -IncludeTotalCount	
Get-ASMailbox -First 1 -Skip 1000|fl	
Get-ASMailbox -UserSid "S-1-5-21-141886343-4149337270-1996687478-6693"	
Get-ASMailbox -MailboxType DisabledUser -MailboxKind OnlinePrimary	

Get-ASUserSyncStatus

SYNOPSIS

Gets Office 365 synchronization.

SYNTAX

Get-ASUserSyncStatus [-Core] <ConnectCore> [-SAMAccountName] <string[]> [-UserPrincipalName] <string[]> [<CommonParameters>]

DESCRIPTION

The Get-ASUserSyncStatus returns last user sync date time. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASUserSyncStatus -UserPrincipalName user1@domain1.onmicrosoft.com Result: Last O365 sync of User: user1 at 04.05.2016 13:09:20 (UTC).	

Set-ASContainer

SYNOPSIS

Sets container.

SYNTAX

Set-ASContainer [-Core] <ConnectCore> [-ContainerId] <Guid[]>[[-DissociateArchiveUser] <SwitchParameter>] [[-RevokeArchiveDissociation] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Set-ASContainer cmdlet sets container parameters. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASContainer -ContainerId 'c2dbaadb-566f-43e4-98ed-030eda591762' -DissociateArchiveUser	/* Command dissociate user from archive container - container becomes onwerless*/
Set-ASContainer -ContainerId 'c2dbaadb-566f-43e4-98ed-030eda591762' -RevokeArchiveDissociation	/* Command enable archive binding to user (if it exists) - container becomes non-ownerless after sync source environment*/

Start-ASSyncADUsers

SYNOPSIS

Start Active Directory sync.

SYNTAX

Start-ASSyncADUsers [-Core] <ConnectCore> [-DistinguishedName] <string> [<CommonParameters>]

DESCRIPTION

The Start-ASSyncADUsers cmdlet starts Active Directory sync. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet

are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASSyncADUsers -DistinguishedName "DC=qa,DC=lab,DC=quadrotech-it,DC=com"	/* Command starts sync of Active Directory users for specified domain */

Start-ASO365SyncMailboxes

SYNOPSIS

Starts active directory sync.

SYNTAX

Start-ASSyncADUsers [-Core] <ConnectCore> [-DistinguishedName] <string[]> [<CommonParameters>]

DESCRIPTION

The Start-ASSyncADUsers cmdlet starts Active Directory sync. After the command is executed, the module status can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASO365SyncMailboxes -LinkId "0d4ad6b3-263c-e611-813b-005056b84ed8" Result: Office365 sync mailboxes command queued.	/* This command is an alternative to the UI Action "Sync Mailboxes" for O365 Link (triggers Office365CollectMailboxes + Office365CollectArchiveMailboxes commands) */

Start-ASADSyncSingleUser

SYNOPSIS

Starts active directory synchronization.

SYNTAX

Start-ASADSyncSingleUser [-Core] <ConnectCore> [-SAMAccountName] <string[]> [-UPN] <string[] > [<CommonParameters>]

DESCRIPTION

The Start-ASADSyncSingleUser starts active directory synchronization. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASADSyncSingleUser -UserPrincipalName user1@l2.lab.quadrotech-it.com Start-ASADSyncSingleUser-SAMAccountName muser1 Start-ASADSyncSingleUser-SAMAccountName user1,maor-l2ex10-qam-1001,user2	/* Syncs single Active Directory User based on UPN or SAMAccountName, (can also enter multiple users per one command) */

Start-ASO365SyncSingleUser

SYNOPSIS

Starts Office 365 synchronization.

SYNTAX

Start-ASO365SyncSingleUser [-Core] <ConnectCore> [-SAMAccountName] <string[]> [-UserPrincipalName] <string[]> [[-CollectMailbox]<SwitchParameter>] [[-SkipUserCheck] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Start-ASO365SyncSingleUser starts Active Directory synchronization. Parameter -SAMAccountName can be partial. If multiple users have a similar name, an exception will occur with the message "Too many users found with userName". You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASO365SyncSingleUser -SAMAccountName user1 Start-ASO365SyncSingleUser -UserPrincipalName user1@domain1.onmicrosoft.com Start-ASO365SyncSingleUser -SAMAccountName user1,user2,user3	/* Sync single Office 365 user based on UPN or SAMAccountName, also it's possible to enter multiple users per one command */ By default, this command queues Office365SyncSingleCloudOnlyUser command to sync single cloud-only user.
Start-ASO365SyncSingleUser -SAMAccountName user1 -CollectMailbox	/* In case -CollectMailbox parameter is used, Office365CollectSingleMailbox and Office365CollectSingleArchiveMailbox commands are queued to collect Office365 user mailboxes */
Start-ASO365SyncSingleUser -UserPrincipalNameuser1@domain1.onmicrosoft.com -SkipUserCheck	/* Syncs single Office 365 user based on UPN without check of existence in Archive Shuttle database */

Add-ASContainerMapping

SYNOPSIS

Creates a new mapping for a source and target container.

SYNTAX

Add-ASContainerMapping [[-Core] <ConnectCore>] -TargetContainerType <TargetContainerTypeEnum> -SourceContainerIds <Guid[]> [-MappingTemplateId <int>] [[-WorkflowPolicyId] <int>] [[-FilterPolicyId] <int>] [-MigrationEnabled] <SwitchParameter> [-ItemGatheringEnabled] <SwitchParameter> [-OwnerlessWorkflowPolicyId] <int> [-IsSameUser] <SwitchParameter> [-IsDifferentUser] <SwitchParameter> [-DifferentUserTargetDomainId] <int> [-DifferentUserMatchingCriteria] <ContainerMappingMatchingFlags[]> [-ContainerMappingAction] <ContainerMappingActionEnum> [-TargetLinkId] <Guid> [-Office365TargetType] <Office365TargetTypeEnum> [-MailboxType]<MailboxKindEnum> [-SecondaryMailboxUnavailableAction] <SecondaryMailboxUnavailableActionEnum> [-Priority] <int> [-PstFileNamePolicyId] <int> [-TargetContainerId] <int> [[-ItemGatheringScheduleDateTime] <DateTime>] [[-MigrationScheduleDateTime] <DateTime>] [[-UserDefinedTokenValue] <string>] [[-UseRecipientsCollectedViaEVIndexForExport] <SwitchParameter>] [[-UseAlternateMailboxes] <SwitchParameter>] [[-CollectSenderRecipientsOnExport] <SwitchParameter>] [[-TargetPathNamePolicyId] <int>] [[-PreMigrationWorkflowPolicyId] <int>] [[-EnableSecondaryMailbox] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Add-ASContainerMapping cmdlet gets SourceContainerId, WorkflowId, FilterPolicyId parameters and creates a mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory except FilterPolicyId.

You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

- All TargetContainerTypes have common parameters:
  - SourceContainerIds, WorkflowPolicyId, FilterPolicyId, MigrationEnabled, ItemGatheringEnabled, OwnerlessWorkflowPolicyId, Priority;
- According to TargetContainerTypes parameter value there are additional parameters.
  - for Enterprise Vault, there are two possibilities:
    i. IsSameUser, ContainerMappingAction, TargetLinkId;
    ii. isDifferentUser, DifferentUserTargetDomainId, DifferentUserMatchingCriteria, ContainerMappingAction, TargetLinkId;
  - for Office 365:
    i. Office365TargetType, MailboxType, SecondaryMailboxUnavailableAction, TargetLinkId;

- for Exchange, there are two possibilities:
  i. MailboxType, SecondaryMailboxUnavailableAction, IsSameUser;
  ii. MailboxType, SecondaryMailboxUnavailableAction, isDifferentUser, DifferentUserTargetDomainId, DifferentUserMatchingCriteria;
- for PST:
  i. TargetLinkId, PstFileNamePolicyId;
- for ProofPoint:
  i. TargetLinkId;
- for Manual (default):
  i. TargetContainerId, SourceContainerIds, WorkflowPolicyId, FilterPolicyId, MigrationEnabled, ItemGatheringEnabled, Priority;

DateTime parameters should be specified in UTC format.

Example:

Add-ASContainerMapping -SourceContainerIds $SourceContainer1.ContainerId -TargetContainerId $TargetContainer1.ContainerId -WorkflowPolicyId 8 -FilterPolicyId 1 -MigrationEnabled -ItemGatheringEnabled	Manual mapping: • New parameter "-TargetContainerType Manual" can be defined • Omit TargetContainerType parameter - command will automatically consider Manual mapping type Parameters supported for manual mapping: -SourceContainerIds -TargetContainerId -WorkflowPolicyId -FilterPolicyId -MigrationEnabled -ItemGatheringEnabled (MigrationEnabled,ItemGatheringEnabled are switch parameters - if used, migration and gathering is enabled)
Add-ASContainerMapping -TargetContainerType Exchange -SourceContainerIds $SourceContainer.ContainerId -MailboxType OnPremisePrimary -IsSameUser -WorkflowPolicyId $wfpolicy.WorkflowPolicyId -FilterPolicyId $filterpolicy.FilterPolicyId -ItemGatheringScheduleDateTime ((GET-DATE).AddDays(1).ToUniversalTime()) -MigrationScheduleDateTime ((GET-DATE).AddDays(1).ToUniversalTime())	/* By specifying highlighted parameters (DateTimeUTC), it is possible to postpone item collection or migration of the mapping. */

Add-ASContainerMapping -TargetContainerType Office365 -SourceContainerIds $SourceContainer.ContainerId -Office365TargetType Normal -MailboxType OnlinePrimary -TargetLinkId "0d4ad6b3-263c-e611-813b-005056b84ed8" -WorkflowPolicyId 1 -OwnerlessWorkflowPolicyId 2 -FilterPolicyId 3 -MigrationEnabled -ItemGatheringEnabled -Priority 99 Command returns result with following information: SourceContainerId : 9e45f73d-1b93-47ce-bcde-6986663c7912 ContainerMappingID : 8 Status : Success ErrorMessage : Error :	Bulk mapping: • Parameter "-TargetContainerType" must be defined first, with desired migration target set (enum available) • Based on the target type, most of the related parameters are dynamically provided. • UserDefinedTokenValue can be specified only for O365 JournalExplosion mapping type
Add-ASContainerMapping -TargetContainerType Office365 -SourceContainerIds $SourceContainer.ContainerId -Office365TargetType Normal -MailboxType OnlinePrimary -TargetLinkId "0d4ad6b3-263c-e611-813b-005056b84ed8" -WorkflowPolicyId 1 -OwnerlessWorkflowPolicyId 2 -FilterPolicyId 3 -ItemGatheringEnabled -Priority 99 -PreMigrationWorkflowPolicyId 107	Bulk mapping: • Same as mapping above but with set -PremigrationWorkflowPolicy • If MigrationEnabled switch is present, than it will be automatically set to false with warning message (WARNING: MigrationEnabled parameter was set to false because migration can't be enabled when Pre-migration policy is set)

Set-ASContainerMapping

SYNOPSIS

Sets attributes of existing container mapping.

SYNTAX

Set-ASContainerMapping [-Core] <ConnectCore> [-ContainerMappingId] <int[]> [[-WorkflowPolicyId] <int>] [[-FilterPolicyId] <int>] [[-MigrationEnabled] <bool>] [[-ItemGatheringEnabled] <bool>] [[-Priority] <byte>] [[-ItemGatheringScheduleDateTime]

<DateTime>] [[-MigrationScheduleDateTime] <DateTime>] [[-ContainerMappingTemplateId] <int?>] [[-UserDefinedTokenValue] <string>] [[-UseRecipientsCollectedViaEVIndexForExport] <bool>] [[-DefaultRolloverBytes] <long?>] [[-DefaultRolloverItemCount] <int?>]

[[-TargetPathNamePolicyId] <int>] [[-PreMigrationWorkflowPolicyId] <int>] [<CommonParameters>]

DESCRIPTION

The Set-ASContainerMapping cmdlet sets various variables of container mapping, After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory except FilterPolicyId. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. DateTime parameters should be specified in UTC format.

Example:

Set-ASContainerMapping -ContainerMappingId 2 -WorkflowPolicyId 6 -FilterPolicyId 1 -Priority 10 -MigrationEnabled 0 -ItemGatheringEnabled 0	/* Sets existing mapping parameters */
Set-ASContainerMapping -ContainerMappingId 1 -Priority 10	/* Sets priority of specified container mapping */
Set-ASContainerMapping -ContainerMappingId 1 -ItemGatheringScheduleDateTime ((GET-DATE).AddDays(1).ToUniversalTime()) -MigrationScheduleDateTime ((GET-DATE).AddDays(1).ToUniversalTime())	/* By specifying highlighted parameters (DateTimeUTC), it is posible to postpone item collection or migration of the mapping (in case it wasn't already enabled) */
Set-ASContainerMapping -ContainerMappingId 1 -ContainerMappingTemplateId 1 Set-ASContainerMapping -ContainerMappingId 1 -ContainerMappingTemplateId 0 Set-ASContainerMapping -ContainerMappingId 2 -UserDefinedTokenValue UDTV	/* Assigns mapping template to specified container mapping */ /* Removes mapping template assignment from specified container mapping */ /* Assigns User defined token value for specified Container Mapping */
Set-ASContainerMapping -ContainerMappingId 1 -DefaultRolloverBytes 524288000 -DefaultRolloverItemCount 600	/* Set DefaultRolloverThreshold bytes and DefaultRolloverItem count for mappings to PST - Threshold will be ignored for mappings with already finished routing */

Set-ASContainerMapping -ContainerMappingId 1 -PreMigrationWorkflowPolicyId 107	/* Set PreMigrationWorkflowPolicy for mapping */

Get-ASBulkMappingWizardTemplate

SYNOPSIS

Gets bulk mapping wizard templates.

SYNTAX

Get-ASBulkMappingWizardTemplate [-Core] <ConnectCore> [[-BulkMappingWizardTemplateId] <int>] [[-BulkMappingWizardTemplateName] <string>] [[-ContainerMappingId] <int>] [<CommonParameters>]

DESCRIPTION

The Get-ASBulkMappingWizardTemplate will return mapping templates. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASBulkMappingWizardTemplate | Where-Obect{$_.Name -like 'Ev2*'}	/* Retrieves all bulk mapping wizard templates templates with name begins with "EV2"*/
Get-ASBulkMappingWizardTemplate -BulkMappingWizardTemplateId 7	/*Retrieves bulk mapping wizard templates template with id 7 */

Get-ASMappings

SYNOPSIS

Displays information about containers mappings.

SYNTAX

Get-ASMappings -Core] <ConnectCore>] [[-ContainerMappingId] <int[]>] [[-ContainerId] <Guid>] [[-ContainerMappingType] <ContainerMappingType>] [[-IncludeVirtualJournalSubmappings] [<SwitchParameter> [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASMappings cmdlet gets data regarding a specified container mapping or mappings. Results contain: Container Mapping Id, Source Container Name, Target Container Name, Item Gathering Enabled, Migration Enabled, Stage 2 Enabled, Stage 2 Finished and VirtualJournalMasterMappingId. By default, all container mapping details are displayed after the

command executes. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASMappings | Select-Object ContainerMappingId,SourceContainer | Sort-Object SourceContainer	/* Retrieves first 1000 mappings, sorted by SourceContainer
Get-ASMappings -ContainerMappingId 1 Get-ASMappings -ContainerMappingId 1,2,3 Get-ASMappings -All	/* Retrieves mapping with exact mapping ID */ /* Retrieves array of mappings based on entered mapping ids */ /* Retrieves all existing mappings */
Get-ASMappings -ContainerId "0c9119c5-1c9a-439b-9067-03f70aeb125a" -ContainerMappingType Source	/* Retrieves mappings which contain specified container ID as a source */
Get-ASMappings -IncludeVirtualJournalSubmappings Get-ASMappings-ContainerMappingId1-IncludeVirtualJournalSubmappings|Where-Object {$_.VirtualJournalMasterContainerMappingId -ne$null }	/* Retrieves also submappings of virtual journal */ /* Retrieves only submappings of specified virtual journal master mapping */

Get-ASPremigrationStatus

SYNOPSIS

Show pre-migration information.

SYNTAX

Get-ASPremigrationStatus [[-Core] <ConnectCore>] [[-ContainerMappingId] <int[]>] [[-All] <int[]>] [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASPremigrationStatus cmdlet gets information about Pre-migration for specified container mappings. By default, all container mapping Pre-migration information is returned after the command executes. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASPremigrationStatus	/* Retrieves pre-migration statuses of all container mappings */
Get-ASPremigrationStatus -ContainerMappingId 1	/* Retrieves pre-migration status of a specific container mapping */
Get-ASPremigrationStatus -First 2 Get-ASPremigrationStatus -Skip 2	/* Retrieves pre-migration status of first 2 container mappings */ /* Skip 2 pre-migration statuses */

Get-ASStage1Statistics

SYNOPSIS

Show Stage 1 statistics.

SYNTAX

Get-ASStage1Statistics -Core] <ConnectCore>] [[-ContainerMappingId] <int[]>] [[-IncludeVirtualJournalSubmappings] [<SwitchParameter> -All] <SwitchParameter>] [[-JournalExplosion] [<SwitchParameter> [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASStage1Statistics cmdlet gets Stage 1 statistics for specified container mappings. Result contains: Container Mapping Id, Source Full Name, IsJournalExplosion, Routed Count, Exported Items Count, Exported Items Percentage, Imported Items Count, Exploded Routings Count *, EnabledForJournalExplosionImport* (* - significant only for JE), Imported Items Percentage and VirtualJournalMasterMappingId. By default, all container mapping statistics are returned after the command executes. When using the JournalExplosion parameter, only JournalExplosion mapping statistics are returned. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASStage1Statistics Get-ASStage1Statistics -ContainerMappingId 1,2,3 Get-ASStage1Statistics -ContainerMappingId "EnterID" *Result:*	

ContainerMappingId	SourceFullName	RoutedCount	ExportStatistics.ProcessedItems	ExportStatistics.ErrorsCount	ExportStatistics.Percentage	ImportStatistics.ProcessedItems	ImportStatistics.ErrorsCount	ImportStatistics.Percentage
1	Archive1	33	32	1	96%	32	0	96%

Get-ASStage1Statistics | fl

Result:

ContainerMappingId : 1

SourceFullName : peho-ms-o365-016

SourceGroup :

RoutedCount : 1000

ExportStatistics : ArchiveShuttle.Common.Resources.PowerShell.Commands.MigrationStatistics

ImportStatistics : ArchiveShuttle.Common.Resources.PowerShell.Commands.MigrationStatistics

Priority : 10

MigrationEnabled : True

ItemGatheringEnabled : True

Stage2Enabled : True

Stage2Finished : 1

Use command with parameter to expand property **ExportStatistics** or **ImportStatistics** (only one property can be expanded at the time)

IsJournalExplosion : False ExplodedRoutingsCount : EnabledForJournalExplosionImport : VirtualJournalMasterContainerMappingId : SourceTag : {tag1, tag2, tag3, tag5} TargetTag : {tag5, tag7} Get-ASStage1Statistics -ContainerMappingId 1 | Select-Object -ExpandProperty ExportStatistics *Result:* ProcessedItems RetryableErrorsCount PermanentErrorsCount ErrorsCount Percentage SizeOriginal (B) SizeComplressed (B) ---------- ---------- ---------- ---------- ---------- ---------- ---------- 33 1 0 0 96	
Commands can be used together with Select/Where clause. Examples below describe how to use it in case of expanded properties where exact property names are required (e.g. **ImportStatistics**.**Percentage**): Get-ASStage1Statistics | Where-Object {$_.ImportStatistics.Percentage -eq 100} Get-ASStage1Statistics \|Where-Object {$_.ImportStatistics.Percentage -le 99} | Select-Object -ExpandProperty ImportStatistics	/* Retrieves only mappings where import progress = 100% */ /* Retrieves only mappings where import progress is less or equal 99% + expand details of these statistics */ /* Retrieves only mappings where export and import statistic values are equal */

Get-ASStage1Statistics | Where-Object {$_.ExportStatistics.ErrorsCount -eq $_.ImportStatistics.ErrorsCount} Get-ASStage1Statistics \| Where-Object {$_.ExportStatistics.ProcessedItems -eq $_.ImportStatistics.ProcessedItems}	
Get-ASStage1Statistics -JournalExplosion	/* Retrieves only journal explosion mappings Journal Explosion has dedicated command to retrieve more statistics: Get-ASJEStage1Statistics */
Get-ASStage1Statistics-IncludeVirtualJournalSubmappings Get-ASStage1Statistics-IncludeVirtualJournalSubmappings|Where-Object {$_.VirtualJournalMasterContainerMappingId -ne$null }	/* Retrieves stats also for submappings of virtual journal master mappings */ /* Retrieves stats only for submappings of specified virtual journal master mappings */

Get-ASStage2Status

SYNOPSIS

Shows Stage 2 information.

SYNTAX

Get-ASStage2Status [[-Core] <ConnectCore>] [[-ContainerMappingId] <int[]>] [[-All] <int[]>] [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASStage2Status cmdlet gets information about Stage 2 for specified container mappings. Result contains: Container Mapping Id, Source Container, Target Container, Current Command Name, Current Command Status, Current Command Error, isFinished and VirtualJournalMasterMappingId. By default, all container mapping Stage 2 information is returned after the command executes. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASStage2Status	

Get-ASStage2Status -All -IncludeTotalCount

Get-ASStage2Status -ContainerMappingId 1,2,3

Get-ASStage2Status -ContainerMappingId 1

Result:

ContainerMappingId	SourceContainer	TargetContainer	CurrentCommandName	CurrentCommandStatus	CurrentCommandError	Finished
---	---	---	---	---	---	---
1	sourceuser1	targetuser1	WaitForImportFinished		Queued	FALSE

Get-ASStage2Status -ContainerMappingId 1 | fl

Result:

ContainerMappingId : 1

SourceContainerName : peho-ms-o365-016

SourceTag : {tag1, tag2, tag3, tag5}

TargetContainerName : peho-ms-o365-016

TargetTag : {tag5, tag7}

WorkflowState : Enabled

CurrentCommandName : NativeImportRename

CurrentCommandId : 1203

CurrentCommandStatus : Success

CurrentCommandError :

CurrentCommandLastUpdateDateUtc : 10.08.2020 13:20:18

CurrentWorkflowRetryFailedAfterMinutes : 60 CurrentWorkflowRetryFailed : True NextCommandName : NextCommandId : IsFinished : True VirtualJournalMasterContainerMappingId :	
Get-ASStage2Status | Where-Object {$_.VirtualJournalMasterContainerMappingId -ne $null }	/* Retrieves only submappings of virtual journal master mappings */

Get-ASWorkflowPolicy

SYNOPSIS

Gets list of available Workflow Policies.

SYNTAX

Get-ASWorkflowPolicy [[-Core] <ConnectCore>] [[-Name] <string[]>] [[-WorkflowPolicyId] <int[]>] [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASWorkflowPolicy cmdlet gets a list of available Workflow Policies. The Workflow Policy can be used as an input parameter for the Add-ASContainerMapping command. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASWorkflowPolicy -IncludeTotalCount

Get-ASWorkflowPolicy -Name "EnterpriseVault to Office 365 Leavers" | fl

Get-ASFilterPolicy

SYNOPSIS

Show list of available Filter Policies.

SYNTAX

Get-ASFilterPolicy [[-Core] <ConnectCore>] [[-Name] <string>] [[-FilterPolicyId] <int[]>] [[-Skip] <uint64>] [[-First] <uint64>] [<CommonParameters>]

DESCRIPTION

The Get-ASFilterPolicy cmdlet gets a list of available Filter Policies. The Filter Policy can be used as an input parameter for the Add-ASContainerMapping command. The core parameter is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASFilterPolicy -IncludeTotalCount

Get-ASFilterPolicy -Name "Default (No Filter)"

Enable-ASCollection

SYNOPSIS

Enables Item Gathering for container.

SYNTAX

Enable-ASCollection [[-Core] <ConnectCore>] [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Enable-ASCollection cmdlet enables item gathering for a specified Container Mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Enable-ASCollection -ContainerMappingId 1

Get-ASMappings| Enable-ASCollection

Result:

Collection was enabled for ContainerMappingId 2

Collection was enabled for ContainerMappingId 3

Collection was enabled for ContainerMappingId 1

Enable-ASMigration

SYNOPSIS

Enables migration for a container.

SYNTAX

Enable-ASMigration [[-Core] <ConnectCore>] [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Enable-ASMigration cmdlet enables migration for a specified Container Mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Enable-ASMigration -ContainerMappingId 1

Get-ASMappings | Enable-ASMigration

Result:

Migration was enabled for ContainerMappingId 1

Migration was enabled for ContainerMappingId 2

Migration was enabled for ContainerMappingId 3

Disable-ASCollection

SYNOPSIS

Disables Item Gathering for a container mapping.

SYNTAX

Disable-ASCollection [[-Core] <ConnectCore>] [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Disable-ASCollection cmdlet disables item gathering for a specified Container Mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Disable-ASCollection -ContainerMappingId 1

Get-ASMappings | Disable-ASCollection

Result:

Collection was disabled for ContainerMappingId 2

Collection was disabled for ContainerMappingId 3

Collection was disabled for ContainerMappingId 1

Disable-ASMigration

SYNOPSIS

Disables migration for container.

SYNTAX

Disable-ASMigration [[-Core] <ConnectCore>] [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Disable-ASMigration cmdlet disables migration for a specified Container Mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Disable-ASMigration -ContainerMappingId 1

Get-ASMappings | Disable-ASMigration

Result:

Migration was disabled for ContainerMappingId 1

Migration was disabled for ContainerMappingId 2

Migration was disabled for ContainerMappingId 3

Enable-ASStage2

SYNOPSIS

Enables Stage 2 migration for container.

SYNTAX

Enable-ASStage2 [[-Core] <ConnectCore>] [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Enable-ASStage2 cmdlet enables Stage 2 for a specified Container Mapping. After the command is executed, the mapping can be checked in the Archive Shuttle user interface. All parameters in the cmdlet are mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Enable-ASStage2 -ContainerMappingId 1

Get-ASMappings | Enable-ASStage2

Result:

Stage2 was enabled for ContainerMappingId 2

Stage2 was enabled for ContainerMappingId 3

Stage2 was enabled for ContainerMappingId 1

Start-ASRetryFailedItems

SYNOPSIS

Starts retry failed items.

SYNTAX

Start-ASRetryFailedItems [-Core] <ConnectCore> [-ContainerMappingId] <int[]> [[-Type] <ImportExportTypeEnum>] [[-RetryableFailed] <SwitchParameter>] [[-PermanentlyFailed] <SwitchParameter>] [[-Hanging] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Start-ASRetryFailedItems command retries failed (retryable, permanent) or hanging items for specified container mapping. This actions can be defined for export, import or both. In case there is no definition within command of what should be retried, then everything available is applied for whole container mapping. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Start-ASRetryFailedItems -ContainerMappingId 1 -Type Export -RetryableFailed -Hanging	/* Retires exported retryable failed and hanging items for specified mapping */
Get-ASMappings -All | Start-ASRetryFailedItems -Type Both	/* Retries all types of exported and imported items (retryable, permanently failed and hanging) for all existing mappings */

Remove-ASContainerMapping

SYNOPSIS

Delete container mapping.

SYNTAX

Remove-ASContainerMapping [-Core] <ConnectCore> [-ContainerMapping] [<int[]>] -IncludeItemData] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Remove-ASContainerMapping cmdlet removes container mapping (migration). Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASContainerMapping -ContainerMappingId 1 Remove-ASContainerMapping -ContainerMappingId 1 -IncludeItemData	/* Deletes specified single mapping */ /* Deletes specified mapping with all related data in Item database */
Remove-ASContainerMapping -ContainerMappingId 1,2,3,4,5 -IncludeItemData Get-ASMappings -First 5 | Remove-ASContainerMapping	/* Deletes specified mappings */ /* First gets list of mappings and then deletes it */

Add-ASMappingConfigurationTemplate

SYNOPSIS

Add new mapping configuration template.

SYNTAX

Add-ASMappingConfigurationTemplate [-Core] <ConnectCore> [-Name] <string> [[-Description] <string>] [<CommonParameters>]

DESCRIPTION

The Add-ASMappingConfigurationTemplate cmdlet creates new Container Mapping Configuration Template and returns it. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASMappingConfigurationTemplate -Name NAME01 -Description DESC01	/* Creates mapping template */

Set-ASMappingConfigurationTemplate

SYNOPSIS

Sets mapping configuration template.

SYNTAX

Set-ASMappingConfigurationTemplate [-Core] <ConnectCore> [-ContainerMappingTemplateId] <int> [[-Name] <string>] [[-Description] <string>] [<CommonParameters>]

DESCRIPTION

The Set-ASMappingConfigurationTemplate cmdlet updates the Container Mapping Configuration Template attributes and returns it. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASMappingConfigurationTemplate -ContainerMappingTemplateId 1 -Name NAME01NEW -Description DESC01NEW	/* Sets new name and description of specified mapping template */
Set configuration properties of mapping template: SEE COMMAND SET-ASCONFIGURATION	

Get-ASMappingConfigurationTemplate

SYNOPSIS

Get mapping configuration template.

SYNTAX

Get-ASMappingConfigurationTemplate [-Core] <ConnectCore> [[-ContainerMappingTemplateId] <int[]>] [[-Name] <string>] [[-Description] <string>] [[-ShowTemplateSettingDefinitions] <bool>] [<CommonParameters>]

DESCRIPTION

The Get-ASMappingConfigurationTemplate cmdlet gets Container Mapping Configuration Template by its ContainerMappingTemplateId or Name. If the parameter is not set, it returns all configuration templates. If ShowTemplateSettingDefinitions is presen,t then the cmdlet returns all setting definitions. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASMappingConfigurationTemplate	/* Gets all existing mapping templates */
Get-ASMappingConfigurationTemplate -ShowTemplateSettingDefinitions | Format-Table	/* Gets details (configuration properties) of all existing mapping templates */ /* Gets mapping template details (configuration properties) of specified container mapping */

Get-ASMappingConfigurationTemplate -ContainerMappingTemplateId 1 -ShowTemplateSettingDefinitions	

Remove-ASMappingConfigurationTemplate

SYNOPSIS

Remove mapping configuration template.

SYNTAX

Remove-ASMappingConfigurationTemplate [-Core] <ConnectCore> [-ContainerMappingTemplateId] <int[]> [<CommonParameters>]

DESCRIPTION

The Remove-ASMappingConfigurationTemplate cmdlet removes Container Mapping Configuration Template. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASMappingConfigurationTemplate -ContainerMappingTemplateId 1 Remove-ASMappingConfigurationTemplate -ContainerMappingTemplateId 2,3	/* Removes specified mapping template */ /* Removes specified mapping templates */
Get-ASMappingConfigurationTemplate | Remove-ASMappingConfigurationTemplate	/* Gets all mapping templates, then removes all */

Add-ASJEUserGroup

SYNOPSIS

Add Archive Shuttle Journal Explosion user group.

SYNTAX

Add-ASJEUserGroup [-Core] <ConnectCore> [-GroupName] [<string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASJEUserGroup cmdlet creates new Archive Shuttle Journal Explosion User groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASJEUserGroup -Name JEGROUP01	
Add-ASJEUserGroup -Name JEGROUP01,JEGROUP02,JEGROUP03,JEGROUP04	

Get-ASJEUserGroup

SYNOPSIS

Get Archive Shuttle Journal Explosion user group.

SYNTAX

Get-ASJEUserGroup [-Core] <ConnectCore> -GroupName] [<string[]>-GroupId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Get-ASJEUserGroup cmdlet gets Archive Shuttle Journal Explosion User groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJEUserGroup -GroupName JEGROUP01 Get-ASJEUserGroup -GroupId	/* Retrieves Journal Explosion group based on its name */ /* Retrieves Journal Explosion group based on its ID */

Set-ASJEUserGroup

SYNOPSIS

Set Archive Shuttle Journal Explosion user group.

SYNTAX

Set-ASJEUserGroup [-Core] <ConnectCore> -GroupName] [<string[]>-GroupId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Set-ASJEUserGroup cmdlet update Archive Shuttle Journal Explosion User groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASJEUserGroup -GroupId 1 -GroupName JEGROUP_01	/* Changes Journal Explosion groups name */

Add-ASFolderNamePolicy

SYNOPSIS

Add Folder Name Policy

SYNTAX

Add-ASFolderNamePolicy [-Core] <ConnectCore> [-Name] <string> -NamingPolicy] [<string>-UsePurgesFolder] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Add-ASFolderNamePolicy cmdlet creates new Folder Name Policy that can be used for Journal Explosion migrations. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and core parameter will be automatically added from session state.

Example:

Add-ASFolderNamePolicy -Name FNP01 -NamingPolicy PREFIX_*originalsmtpaddress*_SUFFIX Add-ASFolderNamePolicy -Name FNP02 -UsePurgesFolder	/* Creates new Folder name policy with specified naming policy */ /* Creates new Folder name policy which points to Purges folder (naming policy does not apply) */

Get-ASFolderNamePolicy

SYNOPSIS

Gets Folder Name policy.

SYNTAX

Get-ASFolderNamePolicy [-Core] <ConnectCore> -Name] [<string>-FolderNamePolicyId] [<int> [<CommonParameters>]

DESCRIPTION

The Get-ASFolderNamePolicy cmdlet gets Folder Name policy information. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. You can use FolderNamePolicyId or Name parameter or both, but using one of them is mandatory.

Example:

Get-ASFolderNamePolicy	/* Retrieves all available Folder Name Policies */
Get-ASFolderNamePolicy -Name FNP01 Get-ASFolderNamePolicy -FolderNamePolicyId 1	/* Retrieves Folder Name Policy with specified name */ /* Retrieves Folder Name Policy with specified ID */

Set-ASFolderNamePolicy

SYNOPSIS

Set Folder Name Policy

SYNTAX

Set-ASFolderNamePolicy [-Core] <ConnectCore> [-FolderNamePolicyId] <int> -Name] [<string>-NamingPolicy] [<string>-UsePurgesFolder] [<SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Set-ASFolderNamePolicy cmdlet sets attribute values for existing Folder Name policy. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Setting Naming Policy and UsePurgesFolder cannot be set together - these parameters are mutually exclusive.

Example:

Set-ASFolderNamePolicy -FolderNamePolicyId 1 -NamingPolicy PREFIX_*originalsmtpaddress*_SUFFIX Set-ASFolderNamePolicy -FolderNamePolicyId 1 -Name FNP_02 -NamingPolicy PREFIX_*originalsmtpaddress*_SUFFIX	/* Changes Naming policy of specified Folder Name Policy */ /* Changes Name and Naming policy of specified Folder Name Policy */

Get-ASJESenderRecipient

SYNOPSIS

Get list of Journal Explosion senderRecipients according to specified parameters.

SYNTAX

Get-ASJESenderRecipient [-Core] <ConnectCore> [-RecipientType] <SenderRecipientEnum> [[-LinkId] <Guid?>] [[-ContainerMappingId] <int[]>] [<CommonParameters>]

DESCRIPTION

The Get-ASJESenderRecipient cmdlet gets list of Journal Explosion senderRecipients. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet has two optional parameters; LinkId and ContainerMappingId, but it is mandatory to use at least one of them.

Example:

Get-ASJESenderRecipient -RecipientType ActiveUser -ContainerMappingId 1 -All -IncludeTotalCount Get-ASJESenderRecipient -RecipientType Leaver -LinkId "ebbd722e-b8b9-4b50-98e8-7ffe95a5fd0d" Get-ASJESenderRecipient -RecipientType ActiveUser -LinkId $JELink.LinkId -All | Sort-Object SenderRecipientId -Unique	/* Retrieves all active SMTP addresses of specified Journal explosion mapping and shows total count of the results (items of the mapping must be exploded to get results) */ /* Retrieves all leaver SMTP addresses of specified Source Link - it combines results if more journal explosion mappings exist for specified link */ /* Retrieves all active SMTP addresses of specified journal explosion Target Link - it combines results if more SourceLinks related to specified JE link (-Unique parameter filters duplicates out) */ /* SenderRecipientId has been removed from Get-ASJESenderRecipient command
Get-ASJESenderRecipient -RecipientType ActiveUser -ContainerMappingId 1 | Where-Object {$_.OriginalAddress -like "*gmail.com"}	/* Retrieves only those active SMTP addresses of specified mapping which meet WHERE condition */
Retrieve only Office 365 leaver senders and recipients $jeO365Links = Get-ASLink -Type Office365JournalExplosion $leaverSenderRecipients = Get-ASJESenderRecipient -RecipientType Leaver -All | Where-Object -Property LinkId -in $jeO365Links.LinkId	/* Gets only Office 365 Journal Explosion links, then retrieval of senders and recipients is based on these links. Due to this, Archive Shuttle can retrieve Office 365 leavers only (Exchange and PST are filtered out) */

Add-ASJEUserMapping

SYNOPSIS

Creates Journal Explosion user mappings.

SYNTAX

Add-ASJEUserMapping [-Core] <ConnectCore> -SenderRecipientMapping] [<PSSenderRecipientForMapping[]>-ContainerMappingId] [<int[]>-SenderRecipientId] [<int[]> [[-OriginalSmtpAddress] <string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASJEUserMapping cmdlet creates Journal Explosion User Mappings. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet has two parameter sets, the first set consists of array of PSSenderRecipientForMapping objects. User can retrieve this array by using Get-ASJESenderRecipient cmdlet. The second set consists of ContainerMappingId array and SenderRecipientId array. When using the second set, ContainerMappingId parameter is mandatory.

Example:

Add-ASJEUserMapping -ContainerMappingId 1	/* Creates Journal Explosion User mappings for all not mapped Senders/Recipients related to specified Container mapping */
Add-ASJEUserMapping -ContainerMappingId 1,2 -SenderRecipientId 1,2 -OriginalSmtpAddress "email@email.com", "email@email.com" Add-ASJEUserMapping -ContainerMappingId 1 -SenderRecipientId 1,2,3 -OriginalSmtpAddress "email@email.com"	/* Creates Journal Explosion User mappings based on pairs as follows: 1 JEUsermapping for [containerMapping 1 - senderRecipient 1] 1 JEUserMapping for [containerMapping 2 - senderRecipient 2] User mappings are created only in case combinations are valid */ /* Creates JournalExplosion User mappings for specified Senders/Recipients related to specified Container mapping */
Get-ASJESenderRecipient -ContainerMappingId 1,2 | Add-ASJEUserMapping $sendRec = Get-ASJESenderRecipient -ContainerMappingId 1 -All \| Where-Object {$_.OriginalAddress -like "*gmail.com"} $sendRec \| Add-ASJEUserMapping \| Format-Table ,$sendRec | Add-ASJEUserMapping | Format-Table #really much faster way to create mappings	/* Creates Journal Explosion User mappings for all not-mapped Senders/Recipients related to specified Container mappings */ /* This procedure shows how to retrieve only subset of Senders/Recipients related to specified container mapping then User mappings can be created through the pipe and there are two ways: 1. Use it as it is and mappings will be created one by one in the background

	2. Convert $sendRec to array by using comma "," and mappings will be created at once, which is much faster way */
Add-ASJEUserMapping -ContainerMappingId 1 -SenderRecipientId 2 -OriginalSmtpAddress "email@email.com" -OwnerlessUser -MappedSmtpAddress email@email.com $ownerless = Get-ASJESenderRecipient -ContainerMappingId 1 -RecipientType ActiveUser\| Where-Object {$null -eq $_.UserSid} $ownerless | ForEach-Object {Add-ASJEUserMapping -ContainerMappingId $_.ContainerMappingId -SenderRecipientId $_.SenderRecipientId -OriginalSmtpAddress $_."email@email.com" -OwnerlessUser -MappedSmtpAddress "email@email.com"}	/* Creates Journal Explosion user mapping for ownerless user. This action is available only for Exchange and PST targets. For Office 365, there is extra set of commandlets (Add-ASJELeaverMapping) */ /* Creates multiple Journal Explosion user mappings with the same mapped address. This is not an issue as it can be then changed by Set-ASJEUserMapping command but once FileName policy is assigned for these mappings, target name will be created accordingly (ignoring mapped address) */

Get-ASJEUserMapping

SYNOPSIS

Get attributes for Journal Explosion user mappings.

SYNTAX

Get-ASJEUserMapping [-Core] <ConnectCore> [[-ContainerMappingId] <int[]>] [[-JournalExplosionUserGroupId] <int[]>] [[-FolderNamePolicyId] <int[]>] [[-All] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASJEUserMapping cmdlet Gets attributes of Journal Explosion User Mappings. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet has one parameter sets, which contains parameters ContainerMappingId, JournalExplosionUserGroupId and FolderNamePolicyId.

Example:

Get-ASJEUserMapping -ContainerMappingId 1 Get-ASJEUserMapping -JournalExplosionUserMappingId 1	/* Retrieves all Journal Explosion User mappings of specified Container mapping */

	/* Retrieves only specified Journal Explosion User mapping */
Get-ASJEUserMapping -ContainerMappingId 1 -JournalExplosionUserGroupId 2	/* Retrieves those Journal Explosion User mappings of specified Container mapping which also belong to specified Journal Explosion Group */

Set-ASJEUserMapping

SYNOPSIS

Set attributes for Journal Explosion user mappings.

SYNTAX

Set-ASJEUserMapping [-Core] <ConnectCore> [-JournalExplosionUserMappingId] <int[]> [[-FileNamePolicyId] <int>] [[-FileNamePolicyName] <string>] [[-RemoveFileNamePolicy] <SwitchParameter>] [[-FolderNamePolicyId] <int>] [[-FolderNamePolicyName] <string>] [[-RemoveFolderNamePolicy] <SwitchParameter>] [[-UserSid] <string>] [[-EnableForImport] <SwitchParameter>] [[-Pause] <SwitchParameter>] [[-Resume] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Set-ASJEUserMapping cmdlet sets attributes of Journal Explosion User Mappings. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet has three parameter sets, the first set contains ResetPolicies parameter, the second contains EnableForImport parameter. The third set consists of JournalExplosionGroupId,JournalExplosionGroupName, FolderNamePolicyId, FolderNamePolicyName and MappedSmtpAddress

Example:

Set-ASJEUserMapping -JournalExplosionUserMappingId 1 -FolderNamePolicyId 2 -JournalExplosionGroupId 1 Set-ASJEUserMapping -JournalExplosionUserMappingId 1 -MappedSMTPAddress maor-l2ex10-001@lab.quadrotech-it.com Set-ASJEUserMapping -JournalExplosionUserMappingId 1 -EnableForImport	/* Assigns specified Folder Name policy and Journal Explotion Group to particular User mapping */ /* Assigns different SMTP address for specified Journal Explosion User mapping */ /* Enables Journal Explosion User mapping for import */ /* Command sets Journal Explosion User mapping to paused state (IsReadyForImport parameter = 2). It can be resumed by other switch parameter -Resume */

Set-ASJEUserMapping -JournalExplosionUserMappingId 1 -Pause Set-ASJEUserMapping -JournalExplosionUserMappingId 1 -EnableToRecreate 1	/* Enables recreation for Journal Explosion User mapping. It is possible to disable recreation also by setting "0" for the same parameter */

Remove-ASJEUserMapping

SYNOPSIS

Removes Journal Explosion user mappings.

SYNTAX

Remove-ASJEUserMapping [-Core] <ConnectCore> -JournalExplosionUserMappingId] [<int[]>-PSJEUserMappingInfo] [<int[]> [<CommonParameters>]

DESCRIPTION

The Remove-ASJEUserMapping cmdlet removes Journal Explosion User Mappings. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet has one parameter; set JournalExplosionUserMappingId, consisting of array of integers. This parameter is mandatory. The user can retrieve this array by using Get-ASJEUserMapping cmdlet.

Example:

Remove-ASJEUserMapping -JournalExplosionUserMappingId 1	/* Removes specified Journal Explosion User mapping */
Get-ASJEUserMapping -ContainerMappingId 1 -All | Remove-ASJEUserMapping | Format-Table	/* Gets all Journal Explosion User mappings of specified Container mapping and removes them */
$JEUserMappings = Get-ASJEUserMapping -ContainerMappingId 1 -All $JEUserMappings \| Remove-ASJEUserMapping \| Format-Table ,$JEUserMappings | Remove-ASJEUserMapping | Format-Table	/* This procedure shows how to retrieveJournal Explosion User mappings of specified Container mapping then User mappings can be removed through the pipe and with two methods: 1. Use it as is and mappings will be removed one by one in the background, or 2. Convert $JEUserMappings to array by using comma "," and mappings will be removed at once, which can be more efficient */

Add-ASJELeaverMapping

SYNOPSIS

Adds new Journal Explosion Leaver Mapping(s).

SYNTAX

Add-ASJELeaverMapping [-Core] <ConnectCore> [[-ContainerMappingId] <int[]>] \[[-SenderRecipientId] <int[]>] [[OriginalSmtpAddress] <string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASJELeaverMapping cmdlet adds new Journal Explosion Leaver Mapping(s). Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASJELeaverMapping -ContainerMappingId 1 Add-ASJELeaverMapping -ContainerMappingId 1 -SenderRecipientId 10 OriginalSmtpAddress "email@email.com"	/* Creates Journal Explosion Leaver mappings based on JE Container Mapping */ /* Creates Journal Explosion Leaver mapping for specified sender/recipient related to specified JE Container Mapping */
Get-ASJESenderRecipient -ContainerMappingId 2 -RecipientType Leaver | Add-ASJELeaverMapping	/* Retrieves only leaver sender/recipients of specified JE Container Mapping and creates Journal Explosion Leaver mappings */

NOTE: After the leaver mapping is created, the result returns property "UserMappingAssociationStatus", which indicates whether there is existing relation to user mapping (Associated/Enabled) or there is no user mapping created yet (not associated).

Get-ASJELeaverMapping

SYNOPSIS

Gets the Journal Explosion Leaver Mapping(s).

SYNTAX

Get-ASJELeaverMapping [-Core] <ConnectCore> [[-ContainerMappingId] <int[]>] [[-JournalExplosionLeaverMappingId] <int[]>] [[-All] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASJELeaverMapping cmdlet gets the Journal Explosion Leaver Mapping(s). Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJELeaverMapping -All -IncludeTotalCount Get-ASJELeaverMapping -ContainerMappingId 2 Get-ASJELeaverMapping -JournalExplosionLeaverMappingId 1,2	/* Retrieves all Journal Explosion Leaver mappings and includes its total count */ /* Retrieves all Journal Explosion Leaver mappings of specified JE Container Mapping */ /* Retrieves only Journal Explosion Leaver mappings based on JournalExplosionLeaverMappingIds*/
$email = "email@domain.com" Get-ASJELeaverMapping -All \| Where-Object {$_.OriginalSmtpAddress -eq $email}	/* Retrieves only leaver mappings with specified original SMTP address */

UserMappingAssociationStatus property:

Not Associated - leaver mapping exists but has no related user mapping created yet

Associated - leaver mapping after "Enable For Import" has existing related user mapping but is not enabled for import (due to incorrect setting e.g. wrong FNP)

Enabled - leaver mapping after "Enable For Import" has existing related user mapping and is enabled for import.

Set-ASJELeaverMapping

SYNOPSIS

Sets the Journal Explosion Leaver Mapping(s) parameters. Runs specific functionality for Journal Explosion Leaver Mappings.

SYNTAX

Set-ASJELeaverMapping [-Core] <ConnectCore> [-JournalExplosionLeaverMappingId] <int[]> [[-FolderNamePolicyId] <int?>] [[-FolderNamePolicyName] <string>] [[-RemoveFolderNamePolicy] <SwitchParameter>] [[-GroupId] <int?>] [[-GroupName] <string>] [[-RemoveGroup] <SwitchParameter>] [[-UserSid] <string>] [[-EnableForImport] <SwitchParameter>] [[-SetAsDistributionList] <SwitchParameter>] [[-MergeByPreviewSMTPAddress] <SwitchParameter>] [[-MergeByJournalExplosionLeaverMappingId] <int?>] [<CommonParameters>]

DESCRIPTION

The Set-ASJELeaverMapping cmdlet sets the Journal Explosion Leaver Mapping(s) parameters. Runs specific functionality for Journal Explosion Leaver Mappings. Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 10 -FolderNamePolicyId 1 Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 10 -RemoveFolderNamePolicy Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 15 -GroupId 1 Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 10 -RemoveGroup Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 10 -SetAsDistributionList Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 14 -EnableForImport Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 14 -UserSid S-1-5-21-141886343-4149337270-1996687478-3137	/* Assigns specified Folder Name Policy to defined Journal Explosion Leaver Mapping */ /* Removes Folder Name Policy from specified Journal Explosion Leaver Mapping */ /* Assigns specified Group to defined Journal Explosion Leaver Mapping */ /* Removes Group from specified Journal Explosion Leaver Mapping */ /* Marks specified Journal Explosion Leaver Mapping as Distribution list */ /* Enables specified Journal Explosion Leaver Mapping for migration */ /* Manual mapping of specified user to specified Journal Explosion Leaver Mapping */
Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 5,6 -MergeByJELeaverMappingId 6 Set-ASJELeaverMapping -JournalExplosionLeaverMappingId 5,6 -MergeByPreviewSMTPAddress Get-ASJELeaverMapping -ContainerMappingId 3 -All | Set-ASJELeaverMapping -MergeByPreviewSMTPAddress	/* Merges specified Journal explosion Leaver Mappings into one target SMTP Address taken from specified Journal Explosion Leaver Mapping */ /* Merges specified Journal Explosion Leaver Mappings based on Preview Address */ /* Merges automatically all Journal Explosion Leaver Mappings related to specified Journal Explosion Container Mapping. As a result, there will be multiple leaver mapping groups based on mutual preview addresses */ /* Merges automatically all Journal Explosion Leaver Mappings. As result there will be

Get-ASJELeaverMapping -All | Set-ASJELeaverMapping -MergeByPreviewSMTPAddress	multiple leaver mapping groups based on mutual preview addresses */ -MergeByPreviewSMTPAddress does not automatically enable mappings for import! IT can be enabled afterwards by running command again with -EnableForImport parameter.

$email = "asqata-leaver-01@outlook.sk"
 	
Get-ASJELeaverMapping -All | Where-Object {$_.OriginalSmtpAddress -eq $email} | ForEach-Object -Begin { $ids = @() } -Process { $ids += $_.JournalExplosionLeaverMappingId } -End { Set-ASJELeaverMapping -JournalExplosionLeaverMappingId $ids -MergeByJELeaverMappingId ($ids | Select-Object -First 1) }
 	
$emails = Get-ASJELeaverMapping -All | Select-Object OriginalSmtpAddress -Unique
 	
ForEach ($email in $emails){
 	
Write-Host "email: $($email.OriginalSmtpAddress)" -ForegroundColor Yellow
 	
$mapping = Get-ASJELeaverMapping -All \| Where-Object {$_.OriginalSmtpAddress -eq $email.OriginalSmtpAddress} | ForEach-Object -Begin { $ids = @() } -Process { $ids += $_.JournalExplosionLeaverMappingId } -End { Set-ASJELeaverMapping -JournalExplosionLeaverMappingId $ids -MergeByJELeaverMappingId ($ids | Select-Object -First 1) }
 	
$mapping | Format-Table -AutoSize
 	
Write-Host
 	
}
	/* Retrieves all Journal Explosion Leaver Mappings with specified original email address and merges it into one mutual target */
	
	/* Retrieves Journal Explosion Leaver Mappings with specified original email address and merges it into one target. This happens in cycle for all original email addresses */

Remove-ASJELeaverMapping

SYNOPSIS

Removes the Journal Explosion Leaver Mapping(s).

SYNTAX

Remove-ASJELeaverMapping [-Core] <ConnectCore> [-JournalExplosionLeaverMappingId] <int[]> [<CommonParameters>]

DESCRIPTION

The Remove-ASJELeaverMapping cmdlet removes the Journal Explosion Leaver Mapping(s). Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASJELeaverMapping -JournalExplosionLeaverMappingId 12 Get-ASJELeaverMapping -ContainerMappingId 2 | Remove-ASJELeaverMapping Get-ASJELeaverMapping -ContainerMappingId 3,4 | Remove-ASJELeaverMapping	/* Removes Journal Explosion Leaver Mapping based on specified ID */ /* Retrieves all Journal Explosion Leaver Mappings of specified Journal Explosion Container Mapping and removes it */ /* Retrieves all Journal Explosion Leaver Mappings of specified Journal Explosion Container Mappings and removes it */ Mappings already enabled for migration cannot be removed!
Get-ASJELeaverMapping -All | Remove-ASJELeaverMapping	/* Retrieves all available Journal Explosion Leaver Mappings in Archive Shuttle database and removes it. Mappings already enabled for migration cannot be removed! */

Get-ASJEStage1Statistics

SYNOPSIS

Get detail statistics for Journal Explosion submappings

SYNTAX

Get-ASJEStage1Statistics [-Core] <ConnectCore> [-ContainerMappingId] <int> [<CommonParameters>]

DESCRIPTION

The Get-ASJEStage1Statistics cmdlet gets Stage1 detail statistics of Journal Explosion User sub-mappings. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJEStage1Statistics -ContainerMappingId 1 -All	/* Retrieves Stage1 statistics of all Journal Explosion User mappings of a specified Container mapping; it is also possible to retrieve stats only for one Container mapping at once (similar to Archive Shuttle UI) */

Add-ASJEEmailAddressRule

SYNOPSIS

Add new Journal Explosion email address rule.

SYNTAX

Add-ASJEEmailAddressRule [-Core] <ConnectCore> [-Expression] <string> [-ContainerMappingId] <int> [-EnableForImport] <SwitchParameter> [<CommonParameters>]

DESCRIPTION

The Add-ASJEEmailAddressRule cmdlet adds new Journal Explosion Email Address Rule. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASJEEmailAddressRule -ContainerMappingId 1 -Expression "*@test.com"	/* Creates new journal explosion email address rule for specified container mapping*/
Add-ASJEEmailAddressRule -ContainerMappingId 1 -Expression "*@test.com" -EnableForImport	/* Creates new journal explosion email address rule for specified container mapping. Each user mapping created by this rule is automatically enabled for import */

Get-ASJEEmailAddressRule

SYNOPSIS

Gets Journal Explosion email address rule.

SYNTAX

Get-ASJEEmailAddressRule [-Core] <ConnectCore> [[-JournalExplosionUserRuleId] <int[]>] [[-ContainerMappingId] <int[]>] [[-Expression] <string[]>] [[-EnableForImport] <bool?>] [<CommonParameters>]

DESCRIPTION

The Get-ASJEEmailAddressRule cmdlet gets Journal Explosion Email Address Rule(s). Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJEEmailAddressRule -ContainerMappingId 1 Get-ASJEEmailAddressRule -JournalExplosionUserRuleId 2 Get-ASJEEmailAddressRule -ContainerMappingId 1 -JournalExplosionUserRuleId 2	/* Gets email address rule for specified container mapping */ /* Gets email address rule with specified rule Id */ /* Gets email address rule for specified container mapping with specified rule id */
Get-ASMappings | Get-ASJEEmailAddressRule | Format-Table Get-ASJEEmailAddressRule | Format-Table	/* Gets all email address rules of all available Journal Explosion mappings */

Set-ASJEEmailAddressRule

SYNOPSIS

Sets Journal Explosion email address rule.

SYNTAX

Set-ASJEEmailAddressRule [-Core] <ConnectCore> [-JournalExplosionUserRuleId] <int> [-Expression] <string> [-EnableForImport] <bool?> [<CommonParameters>]

DESCRIPTION

The Set-ASJEEmailAddressRule cmdlet sets Journal Explosion Email Address rule. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASJEEmailAddressRule -JournalExplosionUserRuleId 1 -Expression "*@TEST001.com" -EnableForImport $false	/* Changes existing email address rule and sets flag EnableForImport to FALSE */

Get-ASMappings | Get-ASJEEmailAddressRule | Set-ASJEEmailAddressRule -EnableForImport $true Get-ASJEEmailAddressRule | Set-ASJEEmailAddressRule -EnableForImport $true	/* Enables all existing email address rules of all available Journal Explosion mappings to TRUE */

Remove-ASJEEmailAddressRule

SYNOPSIS

Removes Journal Explosion email address rule.

SYNTAX

Remove-ASJEEmailAddressRule [-Core] <ConnectCore> [[-JournalExplosionUserRuleId] <int[]>] [[-ContainerMappingId] <int[]>] [[-Expression] <string[]>] [<CommonParameters>]

DESCRIPTION

The Remove-ASJEEmailAddressRule cmdlet removes Journal Explosion Email Address Rule(s). Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Advanced parameters are supported by this command (Confirm,Whatif).

Example:

Remove-ASJEEmailAddressRule -JournalExplosionUserRuleId 1 Remove-ASJEEmailAddressRule -ContainerMappingId 1 Remove-ASJEEmailAddressRule -ContainerMappingId 1 -JournalExplosionUserRuleId 1 Remove-ASJEEmailAddressRule -Expression *@qa.quadrotech-it.com Remove-ASJEEmailAddressRule -ContainerMappingId 1 -Confirm:$false	/* Removes specified email address rule */ /* Remove all email address rules of specified Journal Explosion mapping */ /* Remove specified email address rule of specified Journal Explosion mapping */ /* Remove email address rule of specified expression */ /* Remove all email address rules of specified Journal Explosion mapping straight away by ignoring confirm action */
Get-ASMappings | ForEach-Object {Remove-ASJEEmailAddressRule -ContainerMappingId $_.ContainerMappingId} Get-ASMappings | Get-ASJEEmailAddressRule | ForEach-Object {Remove-ASJEEmailAddressRule -	/* Removes all email address rules of all Journal Explosion mappings */

JournalExplosionUserRuleId $_.JournalExplosionUserRuleId} Remove-ASJEEmailAddressRule	

Add-ASJEADUserRule

SYNOPSIS

Adds Journal Explosion Active Directory User Rule.

SYNTAX

Add-ASJEADUserRule [-Core] <ConnectCore> [-ContainerMappingId] <int> [-UserSid] <string[]> [<CommonParameters>]

DESCRIPTION

The Add-ASJEADUserRule cmdlet creates new Journal Explosion Active Directory User Rule. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASJEADUserRule -ContainerMappingId 1 -UserSid S-1-5-21-141886343-4149337270-1996687478-1135	/* Creates Active Directory user rule for specified Journal Explosion mapping based on various user properties */
Get-ASUser -First 10 | Add-ASJEADUserRule -ContainerMappingId 1	/* Creates ten Active Directory user rules for specified Journal Explosion mapping */

Get-ASJEADUserRule

SYNOPSIS

Gets Journal Explosion Active Directory User Rule.

SYNTAX

Get-ASJEADUserRule [-Core] <ConnectCore> [[-JournalExplosionUserQueueId] <int[]>] [[-ContainerMappingId] <int[]>] [[-UserSid] <string[]>] [[-All] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASJEADUserRule cmdlet gets the Journal Explosion Active Directory User Rule. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJEADUserRule -ContainerMappingId 1 -All Get-ASJEADUserRule -JournalExplosionUserQueueId 1	/* Retrieves all Active Directory user rules of specified Journal Explosion mapping */ /* Retrieves specific Active Directory user rule */
Get-ASMappings | Get-ASJEADUserRule -All | Sort-Object -Property JournalExplosionUserQueueId | Format-Table Get-ASJEADUserRule	/* Retrieves all Active Directory user rules of all Journal Explosion mappings */

Remove-ASJEADUserRule

SYNOPSIS

Removes Journal Explosion Active Directory User Rule.

SYNTAX

Remove-ASJEADUserRule [-Core] <ConnectCore> [[-JournalExplosionUserQueueId] <int[]>] [[-ContainerMappingId] <int[]>] [[-UserSid] <string[]>] [<CommonParameters>]

DESCRIPTION

The Remove-ASJEADUserRule cmdlet removes the Journal Explosion Active Directory User Rule. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASJEADUserRule -ContainerMappingId 1 Remove-ASJEADUserRule -ContainerMappingId 1 -Confirm:$false Remove-ASJEADUserRule -JournalExplosionUserQueueId 22 Remove-ASJEADUserRule -UserSid "S-1-5-21-1234567890-100000000-100000000-1000"	/* Removes all Active Directory user rules of specified Journal Explosion mapping */ /* Removes all Active Directory user rules of specified Journal Explosion mapping straight away by ignoring confirm action */ /* Removes specified Active Directory user rule */ /* Removes Active Directory user rules for specified mailbox based on UserSid */
Get-ASMappings | ForEach-Object {Remove-ASJEADUserRule -ContainerMappingId $_.ContainerMappingId} Get-ASMappings | Get-ASJEADUserRule | ForEach-Object {Remove-ASJEADUserRule -	/* Removes all Active Directory user rules of all Journal Explosion mappings */

JournalExplosionUserQueueId $_.JournalExplosionUserQueueId} Remove-ASJEADUserRule	

Add-ASJEDistributionListRule

SYNOPSIS

Add Distribution List rule.

SYNTAX

Add-ASJEDistributionListRule [-Core] <ConnectCore> [-ContainerMappingId] <int> [-Expression] <string> [<CommonParameters>

DESCRIPTION

The Add-ASJEDistributionListRule will add new distribution list rule. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASJEDistributionListRule -ContainerMappingId 1 -Expression "dll*"	/* Creates Distribution list rule for specified Journal Transformation mapping with specified expression */

Get-ASJEDistributionListRule

SYNOPSIS

Get Distribution List Rule.

SYNTAX

Get-ASJEDistributionListRule [-Core] <ConnectCore> [-JournalExplosionDistributionListRuleId] <int[]> [-ContainerMappingId] <int[]> [-Expression] <string[]> [<CommonParameters>]

DESCRIPTION

The Get-ASJEDistributionListRule will return existing distribution list rules. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASJEDistributionListRule Get-ASJEDistributionListRule -ContainerMappingId 1,2	/* Retrieves all Distribution list rules for all Journal Transformation mapping */

Get-ASJEDistributionListRule -JournalExplosionDistributionListRuleId 1,2 Get-ASJEDistributionListRule -Expression "dll*"	/* Retrieves all Distribution list rules for specified Journal Transformation mappings */ /* Retrieves Distribution list rules of specified IDs */ /* Retrieves Distribution list rule of specified expression */

Remove-ASJEDistributionListRule

SYNOPSIS

Remove Distribution List rule.

SYNTAX

Remove-ASJEDistributionListRule [-Core] <ConnectCore> [-JournalExplosionDistributionListRuleId] <int[]> [-ContainerMappingId] <int[]> [-Expression] <string[]> [<CommonParameters>]

DESCRIPTION

The Remove-ASJEDistributionListRule will remove existing distribution list rule. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Remove-ASJEDistributionListRule Remove-ASJEDistributionListRule -ContainerMappingId 1,2 -Confirm:$false Remove-ASJEDistributionListRule -JournalExplosionDistributionListRuleId 1,2 Remove-ASJEDistributionListRule -Expression "dll*"	/* Removes Distribution list rules for all Journal Transformation mappings */ /* Removes Distribution list rules for specified Journal Transformation mappings by ignoring confirm action */ /* Removes Distribution list rules for specified Journal Transformation mappings by ignoring confirm action */ /* Removes Distribution list rule of specified expression */

Set-ASJEDistributionListRule

SYNOPSIS

Set Distribution List rule.

SYNTAX

Set-ASJEDistributionListRule [-Core] <ConnectCore> [-JournalExplosionDistributionListRuleId] <int> [-Expression] <string> [<CommonParameters>]

DESCRIPTION

The Set-ASJEDistributionListRule will adjust existing distribution list rule. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASJEDistributionListRule -JournalExplosionDistributionListRuleId 1 -Expression "dll2*"	/* Changes Distribution list rule expression for specified Journal Transformation DistributionListRuleId */

Copy-ASJEAutoCreationRules (Copy-ASJECreationRules)

SYNOPSIS

Copies Journal Explosion Auto Creation rule(s).

SYNTAX

Copy-ASJEAutoCreationRules [-Core] <ConnectCore> [-SourceContainerMappingId] <int> [-TargetContainerMappingIds] <int[]> [[-EmailAddressRule] <SwitchParameter>] [[-UserQueue] <SwitchParameter>] [[-DistributionListRule] <SwitchParameter>] [[-ReplaceExistingRules] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Copy-ASJECreationRules cmdlet copies the Journal Explosion Creation rule(s). Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Copy-ASJEAutoCreationRules -TargetContainerMappingIds 2,3 -SourceContainerMappingId 1 -EmailAddressRule -UserQueue -DistributionListRule Copy-ASJEAutoCreationRules -SourceContainerMappingId 1 -TargetContainerMappingIds 2,3 -EmailAddressRule -UserQueue -DistributionListRule -ReplaceExistingRules Copy-ASJEAutoCreationRules -SourceJournalExplosionEmailRuleIds 119,120,121 -SourceJournalExplosionUserQueueIds	/* Copies all Auto-creation rules from specified source Journal Explosion mapping into specified Journal Explosion mappings (origin rules in target mappings will be kept */ /* Copies all auto-creation rules from specified source Journal Explosion mapping into specified mappings (origin rules in target mappings will be removed) */ /* Copies specified email address and Active Directory user rules into specified Journal Explosion mappings */ /* Copy selected auto-creation rules (EmailAddressRule and DistributionListRule) from specified source Journal Transformation

351,352,353 -TargetContainerMappingIds 2,3 -ReplaceExistingRules Copy-ASJEAutoCreationRules -TargetContainerMappingIds 2,3 -SourceContainerMappingId 1 -EmailAddressRule -DistributionListRule Copy-ASJEAutoCreationRules -TargetContainerMappingIds 2,3 -SourceContainerMappingId 1 -EmailAddressRule -DistributionListRule	mapping into specified mappings (origin rules in target mappings will be kept */ /* Copy selected auto-creation rules (EmailAddressRule and DistributionListRule) from specified source Journal Transformation mapping into specified mappings (only selected rules in target mappings will be removed, for other rules - in this case -UserQueue rules will be kept */

Get-ASMailboxQuotaStatus

SYNOPSIS

Gets the Mailbox Quota status.

SYNTAX

Get-ASMailboxQuotaStatus [-Core] <ConnectCore> [[-UserSid] <string[]>] [[-PrimarySmtpAddress] <string[]>] [[-JournalTransformation] <SwitchParameter>] [[-All] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASMailboxQuotaStatus cmdlet gets the Mailbox Quota status. Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASMailboxQuotaStatus -UserSid "S-1-5-21-1234567890-100000000-100000000-1000" Get-ASMailboxQuotaStatus -PrimarySmtpAddress "email@email.com"	/* Retrieves storage quotas for specified mailbox based on UserSID */ /* Retrieves storage quotas for specified mailbox based on email address */
Get-ASMailboxQuotaStatus Get-ASMailboxQuotaStatus -JournalTransformation Get-ASMailboxQuotaStatus -All -IncludeTotalCount	/* Retrieves storage quotas for first 1000 mailboxes */ /* Retrieves storage quotas only for Journal Transformation user mappings */ /* Retrieves storage quotas for all mailboxes and includes count of returned records */

Update-ASMailboxSizeAndQuota

SYNOPSIS

Forces mailbox size and quota check.

SYNTAX

Update-ASMailboxSizeAndQuota [-Core] <ConnectCore> [[-UserSid] <string[]>] [[-PrimarySmtpAddress] <string[]>] [[-JournalTransformation]<SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Update-ASMailboxSizeAndQuota cmdlet forces mailbox size and quota check. Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Update-ASMailboxSizeAndQuota -UserSid "S-1-5-21-1234567890-100000000-100000000-1000" Update-ASMailboxSizeAndQuota -PrimarySmtpAddress "email@email.com" Update-ASMailboxSizeAndQuota -PrimarySmtpAddress "email@email.com" -JournalTransformation	/* Sends update command for specified mailbox based on UserSid*/ /* Sends update command for specified mailbox based on email address */ /* Sends update command for specified mailbox based on email address and JT switch*/
Get-ASMailboxQuotaStatus -All | Update-ASMailboxSizeAndQuota -UserSid $_.UserSid	/* First retrieves all mailboxes with quota records, then sends update command for all of them*/

Get-ASJELeaversStatus

SYNOPSIS

Get list of Journal Explosion leavers enabled for finalization according to specified parameters.

SYNTAX

Get-ASJELeaversStatus [[-UserSids] <string[]>] [[-PrimarySmtpAddress] <string[]>] [[-All] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASJELeaversStatus cmdlet gets a list of Journal Explosion leavers enabled for finalization. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Cmdlet

has two optional parameters; LinkId and ContainerMappingId, but it is mandatory to use at least one of them.

Example:

Get-ASJELeaversStatus -All -IncludeTotalCount Get-ASJELeaversStatus -UserSids S-1-5-21-1234567890-100000000-100000000-1059 Get-ASJELeaversStatus -PrimarySmtpAddresses youruser@yourdomain.com	/* Retrieves all available Journal Explosion leavers enabled for finalization and adds also total count of result*/ /* Retrieves specific Journal Explosion leaver based on UserSid */ /* Retrieves specific Journal Explosion leaver based on PrimarySMTP address */
Get-ASJELeaversStatus -UserSids S-1-5-21-1234567890-100000000-100000000-1059,S-1-5-21-1234567890-100000000-100000000-1058 Get-ASJELeaversStatus -PrimarySmtpAddresses youruser1@yourdomain1.com,youruser2@yourdomain2.com	/* Retrieves specific Journal Explosion leaver based on UserSid list */ /* Retrieves specific Journal Explosion leaver based on PrimarySMTP address list */

Set-ASJELeaversStatus

SYNOPSIS

Sets the Journal Explosion Leaver Status parameters. Runs specific functionality for Journal Explosion Leaver Status.

SYNTAX

Set-ASJELeaversStatus [-Core] <ConnectCore> [-UserSid] [<string[]>] -PrimarySmtpAddress] [<string[]> [-Action] <LeaverStatusAction> [<CommonParameters>]

DESCRIPTION

The Set-ASJELeaversStatus cmdlet sets the Journal Explosion Leaver Status parameters. Runs specific functionality for Journal Explosion Leaver Status. Add Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASJELeaversStatus -PrimarySmtpAddresses "youruser@yourdomain.com" -Action ConvertToUserMailbox	/* Converts single shared mailbox into normal mailbox based on email address */ /* Retries failed finalization step for single user based on UserSid */

Set-ASJELeaversStatus -UserSids S-1-5-21-1234567890-100000000-100000000-1046 -Action RetryFailedStep Set-ASJELeaversStatus -UserSids S-1-5-21-1234567890-100000000-100000000-1046 -Action Finalize	/* Starts finalization process of specified Journal Explosion user */
Set-ASJELeaversStatus -UserSids (Get-ASJELeaversStatus).UserSid -RetryFailedStep	/* Retries failed steps of multiple Journal Explosion leavers */
Get-ASJELeaversStatus | ForEach-Object -Begin { $ids = @()} -Process { $ids += $_.UserSid} -End { Set-ASJELeaversStatus -UserSids $ids -Action RetryFailedStep} Get-ASJELeaversStatus | ForEach-Object {Set-ASJELeaversStatus -UserSids $_.UserSid -Action RetryFailedStep} Get-ASJELeaversStatus | ForEach-Object {Set-ASJELeaversStatus -UserSids $_.UserSid -Action ConvertToUserMailbox} Get-ASJELeaversStatus | ForEach-Object -Begin { $ids = @()} -Process { $ids += $_.UserSid} -End { Set-ASJELeaversStatus -UserSids $ids -Action ConvertToUserMailbox}	/* Examples of how to use actions via pipes: */

Add-ASGroup

SYNOPSIS

Add Archive Shuttle group.

SYNTAX

Add-ASGroup [-Core] <ConnectCore> [-GroupName] [<string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASGroup cmdlet gets Archive Shuttle user groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASGroup -Name GROUP01	
Add-ASGroup -Name GROUP01,GROUP02,GROUP03	

Set-ASGroup

SYNOPSIS

Set Archive Shuttle group.

SYNTAX

Set-ASGroup [-Core] <ConnectCore> -GroupName] [<string[]>-GroupId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Get-ASGroup cmdlet update Archive Shuttle groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASGroup -GroupName NEWGROUPNAME -GroupId 1	

Get-ASGroup

SYNOPSIS

Get Archive Shuttle group.

SYNTAX

Get-ASGroup [-Core] <ConnectCore> -GroupName] [<string[]>-GroupId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Get-ASGroup cmdlet gets Archive Shuttle groups. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASGroup -GroupName GROUP01	

Add-ASUserToGroup

SYNOPSIS

Add Archive Shuttle user to group.

SYNTAX

Add-ASUserToGroup [-Core] <ConnectCore> [-UserSid] [<SecurityIdentifier[]>] [-SAMAccountName] [<string[]>] -GroupName] [<string> [<CommonParameters>]

DESCRIPTION

The Add-ASUserToGroup cmdlet adds a user to a group. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASUserToGroup -SAMAccountName userSAname -GroupName GROUP01 Add-ASUserToGroup -SAMAccountName domainName\userSAname -GroupName GROUP01	
CSV Import CSV file contains multiple rows with data (UserSid, SAMAccountName,GroupName). Because of this, different users can be assigned into different groups. In the case of SAMAccountName, domains can also be used (domainname\samaccountname): Import-Csv -Path "\\Server01\Folder01\group.csv" | ForEach-Object { Add-ASUserToGroup -UserSid $_.UserSid -GroupName $_.GroupName} Import-Csv -Path "\\Server01\Folder01\group2.csv" | ForEach-Object { Add-ASUserToGroup -SAMAccountName $_.SAMAccountName -GroupName $_.GroupName} CSV file only contains a list of users (either with UserSID or SAMAccountName). The names of the rows (headers) must be exactly **UserSid** or **SAMAccountName**. With this approach, the entire list of users within file can be assigned to one group only: Import-Csv -Path "\\Server01\Folder01\group3.csv" | Add-ASUserToGroup -GroupName GROUP01	

Add-ASTag

SYNOPSIS

Add Archive Shuttle Tag.

SYNTAX

Add-ASTag [-Core] <ConnectCore> [-TagName] [<string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASTag cmdlet creates Archive Shuttle Tags. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Add-ASTag -TagName TAG01 Add-ASTag -TagName TAG01,TAG02,TAG03	

Set-ASTag

SYNOPSIS

Set Archive Shuttle tag.

SYNTAX

Set-ASTag [-Core] <ConnectCore> -TagName] [<string[]>-TagId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Get-ASTag cmdlet update Archive Shuttle tags. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASTag -TagName NEWTAGNAME -TagId 1	

Get-ASTag

SYNOPSIS

Get Archive Shuttle tag.

SYNTAX

Get-ASTag [-Core] <ConnectCore> -TagName] [<string[]>-TagId] [<int[]> [<CommonParameters>]

DESCRIPTION

The Get-ASTag cmdlet gets Archive Shuttle tags. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASTag -TagName TAG01	

Add-ASContainerToTag

SYNOPSIS

Add Archive Shuttle container to tag.

SYNTAX

Add-ASContainerToTag [-Core] <ConnectCore> [[-ContainerId] <Guid[]>] [[-SAMAccountName] <string[]>] [[-UserSid] <SecurityIdentifier[]>] [-TagName] <string[]> [[-TagId] <int[]>] [[-GroupName] <string[]>] [<CommonParameters>]

DESCRIPTION

The Add-ASContainerToTag cmdlet adds a container to a group. Core parameter in the cmdlet is mandatory. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Assign 2 containers to specific Tag:

Add-ASContainerToTag -ContainerId ID01,ID02 -TagName TAG01

Assign container to multiple Tags

Add-ASContainerToTag -ContainerId ID01 -TagName TAG01,TAG02

Assign all users from specified Group into defined Tag:

Add-ASContainerToTag -GroupName GROUP01 -TagName TAG01

Assign specified containers + all users from specified Group to defined Tag

Add-ASContainerToTag -ContainerId ID01 -TagName TAG01 -GroupName GROUP01

CSV Import

CSV file contains multiple rows with data (Containerid, TagName). Because of this, different users can be assigned into different groups:

Import-Csv -Path "\\Server01\Folder01\tag.csv" | ForEach-Object { Add-ASContainerToTag -Containerid $_.Containerid -TagName $_.TagName}

(CSV should contain **SAMAccountName** or **UserSid** in first column, the rest of the columns are considered as Tag names)

To get a list of all containers assigned to specific Tag, see Get-ASContainers command.

Special parameters

- Each one command also supports the PowerShell paging functionality. Get-ASContainers -core $core -first 10 -skip 0
  - By default, paging is set to 1000 results (to override this, some of the commands have the -All parameter available)
- If you want to add output to file then use this command: Get-ASContainers -core $core |Out-File C:\ContainersList.txt
- PowerShell's help for each command is supportedworks.
- Syntax: help-ASpowershell command (other example): Get-Help "Set-ASModuleLogLevel"
- All parameters ignore paging and throws all results (available for specific commands only)
- IncludeTotalCount parameter also returns total count of results regardless of the applied filter; the total count is always thrown

Recreate-ASMapiProfile

SYNOPSIS

Re-creates MAPI profile for the provided module.

SYNTAX

Recreate-ASMapiProfile [-Core] <ConnectCore> [-ModuleId] <Guid> [<CommonParameters>]

DESCRIPTION

The Recreate-ASMapiProfile cmdlet re-creates MAPI profile for the provided module. You need to run the PowerShell session as an administrator to have rights to access and do changes in registry. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Recreate-ASMapiProfile -ModuleId 'A39E0B88-DB8D-EA11-81A5-005056B84ED8'	/* Recreate MAPI profile is sent to NativeFormatImport module, which will

	delete all profiles and recreate MAPI profile. */

Set-ASStage2Status

SYNOPSIS

Set the status for container mapping in stage 2.

SYNTAX

Set-ASStage2Status [-Core] <ConnectCore> [-ContainerMappingId] <int[]> [-Action] <Stage2Action> [[-WorkflowPolicyId] <int>] [[-WorkflowPolicyName] <string>] [<CommonParameters>]

DESCRIPTION

The Set-ASStage2Action will run the chosen stage 2 action. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASStage2Status -ContainerMappingId '2' -Action ChangePolicy -WorkflowPolicyId 145 Set-ASStage2Status -ContainerMappingId '2' -Action Retry	

Set-ASPremigrationAction

SYNOPSIS

Set the status for container mapping in pre-migration.

SYNTAX

Set-ASPremigrationAction [-Core] <ConnectCore> [-ContainerMappingId] <int[]> [-Action] <Stage2Action> [[-WorkflowPolicyId] <int>] [[-WorkflowPolicyName] <string>] [<CommonParameters>]

DESCRIPTION

The Set-ASPremigrationAction will run choosen premigration action. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASPremigrationAction -	/* Will change pre-migration workflow policy

ContainerMappingId 3 -Action ChangePolicy -WorkflowPolicyId 107	for specified ContainerMappingId */
Set-ASPremigrationAction -ContainerMappingId 3 -Action Pause	/* Will pause pre-migration workflow policy */

Get-ASHealthStatus

SYNOPSIS

Get the health status (UI/WS/DB/License) of Archive Shuttle/MM.

SYNTAX

Get-ASHealthStatus [-Core] <ConnectCore> [[-UIStatus] <SwitchParameter>] [[-WSStatus] <SwitchParameter>] [[-DBStatus] <SwitchParameter>] [[-IncludeItemDbs] <SwitchParameter>] [[-LicenseStatus] <SwitchParameter>] [<CommonParameters>]

DESCRIPTION

The Get-ASHealthStatus will return the AS/MM health status for UI/WS/DB/License. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state. Output is in the form of objects that can be further parsed in the pipeline. The user can use command ConvertToJson to unpack the object to a .json format. In the output, there is also Boolean object with name "IsHealthy" that is set to true when no errors/exceptions occur during the check, and also object errors where the found errors are stored and shown when some exceptions occur.

Example:

Get-ASHealthStatus	/* Retrieve all parameters */
Get-ASHealthStatus | ConvertTo-Json -Depth 6	/* Retrieve all parameters unpacked*/

Get-ASModuleCredential

SYNOPSIS

Gets module credentials.

SYNTAX

Get-ASModuleCredential [-Core] <ConnectCore> [-CredentialType] <CredentialType> [[-Name] <string>] [[-ModuleId] <Guid>] <CommonParameters>]

DESCRIPTION

The Get-ASModuleCredential cmdlet gets module credential.You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Get-ASModuleCredential -CredentialType Office365 Get-ASModuleCredential -CredentialType Exchange | Out-File 'c:\check.txt'	/* Retrieve all credentials (type Office 365) */ /* Retrieve all credentials (type Exchange) and saves it into a .txt file*/

Set-ASModuleCredential

SYNOPSIS

Sets module credentials.

SYNTAX

Set-ASModuleCredential [-Core] <ConnectCore> [-ModuleCredentialId] <int> [-IsActive] <bool> [<CommonParameters>]

DESCRIPTION

The Set-ASModuleCredential cmdlet sets module credential properties. You can specify core as a parameter, or call Connect-ASCore once and the core parameter will be automatically added from the session state.

Example:

Set-ASModuleCredential -ModuleCredentialId 59 -IsActive 1 Get-ASModuleCredential -CredentialType Office365 | Set-ASModuleCredential -IsActive 0	/* Sets "active" on true for modulecredentialid 59 */ /* Retrieve all credentials (type Office 365) and disables all of these types*/