

UPA PowerShell Tools

Universal Policy Administrator — PowerShell Administrator Guide

UPA PowerShell Tools brings Universal Policy Administrator to the command line. Everything you do in the console — finding a policy, checking it out, submitting it for approval, reviewing who changed what — has a matching cmdlet, so you can script the routine work, report across hundreds of policies in one pass, and fit UPA into the automation you already run.

The cmdlets act as the account you sign in with. What you can do from PowerShell is governed by the same Roles, Views and Assignments as the console, and everything you change is recorded in the same audit trail. This guide covers installation, connecting, the tasks administrators reach for most, and a reference for all 189 cmdlets.

Before you begin

You need	Details
Windows PowerShell 5.1	The tools are a Windows PowerShell snap-in. Use <code>powershell.exe</code> — they do not load in PowerShell 7 (<code>pwsh.exe</code>).
.NET Framework 4.8	Included with current releases of Windows Server and Windows.
Local administrator rights	Needed to install. The tools install for every user of the machine.
HTTPS access to your UPA server	The same address you use to open the console, for example <code>https://upa.contoso.com</code> .
A UPA account	Entered in UPN form, for example <code>admin@contoso.com</code> .

Install the tools on any Windows machine that can reach your UPA server — the UPA server itself, or an administrator's workstation.

Step 1 — Get the installer

Ask your Full Armor contact for **UPA PowerShell Tools.msi**.

If the file arrived through a browser or by email, Windows marks it as downloaded and may refuse to run it. Clear the mark first: right-click the file, choose **Properties**, tick **Unblock** and click **OK** — or from PowerShell:

```
Unblock-File -Path '.\UPA PowerShell Tools.msi'
```

Step 2 — Install

Double-click **UPA PowerShell Tools.msi** and follow the prompts, or install silently from an elevated prompt:

```
msiexec /i "UPA PowerShell Tools.msi" /qn
```

The tools install to `C:\Program Files\FullArmor\AD Bridge\Powershell Snapin` and appear in **Apps & features** as **UPA Powershell Tools**. To confirm the snap-in is registered:

```
Get-PSSnapin -Registered -Name UPA.Management
```

Step 3 — Load the snap-in

Open **Windows PowerShell** — not PowerShell 7 — and add the snap-in to your session:

```
Add-PSSnapin UPA.Management
Get-Command -PSSnapin UPA.Management | Measure-Object      # Count : 189
```

To load it automatically in every session, add it to your PowerShell profile:

```
if (-not (Test-Path $PROFILE)) { New-Item -ItemType File -Path $PROFILE -Force | Out-Null }
Add-Content -Path $PROFILE -Value 'Add-PSSnapin UPA.Management -ErrorAction SilentlyContinue'
```

Step 4 — Connect to your UPA server

Every session starts with `Get-HAPICConnection`. Give it your credentials and the address of your UPA server:

```
$cred = Get-Credential -Message 'UPA account, e.g. admin@contoso.com'
$cred | Get-HAPICConnection -HAPIServerUrl 'https://upa.contoso.com'
```

Then confirm the server is responding. Each check returns a response time and **Good**:

```
Test-GatekeeperHealth -HAPIServerUrl 'https://upa.contoso.com'      # 8ms Good
Test-GatewayHealth    -HAPIServerUrl 'https://upa.contoso.com'      # 10ms Good
```

If your organization signs in to UPA through a SAML identity provider, name the provider instead of passing a credential:

```
Get-HAPICConnection -HAPIServerUrl 'https://upa.contoso.com' -SAMLProvider 'Contoso SSO'
```

Connect at the top level of your session or script. A connection belongs to the scope that opens it. If you call `Get-HAPICConnection` inside a function, the connection ends when the function returns and every later command fails. Connect once, at the top, and any function you call afterwards can use it.

Certificates. Your machine must trust the certificate your UPA server presents. In a test environment that uses a self-signed certificate, add `-SkipSslCertificateVerification` to `Get-HAPICConnection`. Don't use it in production.

Step 5 — What your account can do

The cmdlets act as the account you connected with, and see exactly what the console shows that account. With the roles described in the *Universal Policy Administrator — Delegation Guide**, the same command returns different results for each person:

Signed in as	Get-UniversalPolicy returns	Get-Roles
Full Administrator	every policy	allowed
Editor, scoped to a Delegation OU	only the policies linked into that OU	not permitted
Approver, scoped to a Delegation OU	only the policies linked into that OU	not permitted
Reviewer	every policy	allowed, read only

To see what your own account holds:

```
Get-RolesForUser | Select-Object Name, @{ n = 'Role'; e = { $_.Role.Name } }
```

Common tasks

These examples assume you have loaded the snap-in and connected.

List and find Universal Policies

```
# every policy you can see, with its current state
Get-UniversalPolicy |
  Select-Object Name, Version, ReleaseVersion,
  @{ n = 'Status'; e = { $_.Status.CurrentStatus } }

# policies whose names match a pattern
Get-UniversalPolicy | Where-Object { $_.Name -like 'CORP-*' }
```

To work with one policy, pick it out by name and keep it in a variable. The later examples use `$up` :

```
$up = @(Get-UniversalPolicy | Where-Object { $_.Name -eq 'CORP-Workstation-Hardening' } )[0]
```

Properties you will use most:

Property	Holds
Name	the policy's name
GlobalPolicyId	its identifier — most cmdlets take this as <code>-UPId</code>
Status.CurrentStatus	where it is in change control (see below)
Version , ReleaseVersion	the current version, and the version last approved

Property	Holds
NativePolicyName	the GPO it is backed by, as domain::name
DomainName , LastModified	its domain, and when it last changed

Get-UniversalPolicy -AllDetails -UPId \$up.GlobalPolicyId returns the policy with its full detail.

Check whether a policy matches its GPO

```
Update-UPGPOSyncStatus -UPId $up.GlobalPolicyId

# InSync ReleaseOutOfSync
# -----
#   True           False
```

To list every policy that has drifted from its GPO:

```
Get-UniversalPolicy | ForEach-Object {
    $s = Update-UPGPOSyncStatus -UPId $_.GlobalPolicyId
    [pscustomobject]@{ Name = $_.Name; InSync = $s.InSync; ReleaseOutOfSync = $s.ReleaseOutOfSync }
} | Where-Object { -not $_.InSync }
```

Each check contacts Active Directory, so allow roughly a minute and a half for every 150 policies.

Compare a policy with its GPO in Active Directory

```
$diff = Get-DiffReport -UPId $up.GlobalPolicyId -VersusGPOInAD
$diff.Changes.Count      # settings that differ from the GPO
$diff.Matches.Count      # settings that match it
```

Move a change through approval

A Universal Policy moves through these states, read from Status.CurrentStatus :

State	Meaning
CheckedOut	being edited, and locked to the account that checked it out
CheckedIn	changes saved, not yet submitted
WaitingForApproval	submitted, awaiting a decision — shown as Pending Approval in the console
Unapproved	rejected by an approver and returned to the author
Approved	released, and ready to publish to Active Directory

```
(Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId).Status.CurrentStatus
```

As an Editor — check out, make your changes, then check in and submit. Both comments are recorded in the policy's history:

```

Checkout-UniversalPolicy -UPId $up.GlobalPolicyId
# ... make your changes - in the console, or from PowerShell as shown below ...
Checkin-UniversalPolicy -UPId $up.GlobalPolicyId -Comment 'Turned off Store app lookup. CR-2041.'
Submit-UniversalPolicy -UPId $up.GlobalPolicyId -Comment 'Ready for review - CR-2041.'

```

Changing registry settings from PowerShell. `Add-GlobalPolicySetting` changes the policy object in your session. Save it with `Update-UniversalPolicy` before you check in, or there is nothing to check in.

Registry settings live in two sections of a Universal Policy: **Computer Configuration** (the section whose `MachinePolicy` is `True`) and **User Configuration** (`MachinePolicy` is `False`). Name the section with `-SectionId`. If you leave it out, the setting may land in User Configuration, whatever the key. A policy created with `New-UniversalPolicy` has no sections until you add the first setting, so this example creates the Computer section when the policy doesn't have one yet:

```

$u = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId # the policy must be checked out
$pol = @($u.Policies)[0]
$sec = if ($pol) { $pol.Sections | Where-Object { $_.PolicyProviderType -eq 'Registry' -and $_.MachinePolicy } |
    Select-Object -First 1 }
if (-not $sec) { # no Computer Configuration section yet - add one
    $sec = New-Object HAPI.SharedLibrary.GlobalPolicy.RegistryPSPProvider
    $sec.SectionId = [guid]::NewGuid(); $sec.MachinePolicy = $true; $sec.Name = 'Windows Machine Registry Settings'
    if (-not $pol) {
        $pol = New-Object HAPI.SharedLibrary.GlobalPolicy.Policy
        $pol.PolicyId = [guid]::NewGuid(); $pol.PolicyType = 'GPO'; $pol.Sections = @($sec)
        $u.Policies = @($pol)
    } else { $pol.Sections = @($pol.Sections) + $sec }
}

Add-GlobalPolicySetting -GP $u -Provider Registry -PolicyId $pol.PolicyId -SectionId $sec.SectionId `
    -Key 'Software\Policies\Microsoft\Windows\Explorer' -ValueName 'NoUseStoreOpenWith' -Value 1 -ValueType DWord |
Out-Null
Update-UniversalPolicy -ThePolicy $u | Out-Null

```

The key is written without a hive (`Software\...`, not `HKLM\Software\...`): the section decides whether it goes to `HKEY_LOCAL_MACHINE` or `HKEY_CURRENT_USER`. For a User Configuration setting, select the section whose `MachinePolicy` is `False` instead.

To abandon a checkout and discard the changes, use `Revert-PolicyCheckout -UPId $up.GlobalPolicyId`.

As an Approver — approve, or reject with `-Approved $false`:

```

Approve-UniversalPolicy -UPId $up.GlobalPolicyId -Approved $true -Comment 'Reviewed against CR-2041.'
Approve-UniversalPolicy -UPId $up.GlobalPolicyId -Approved $false -Comment 'Needs a rollback plan.'

```

Publishing. Once a policy is approved, publish it to Active Directory — see the next task.

Check the result. Several of these cmdlets report a failure in the object they return rather than by stopping your script. Where the result has a `Success` property, test it — and read `Error` for the reason. Otherwise, read the policy's status afterwards.

```
$r = Checkin-UniversalPolicy -UPId $up.GlobalPolicyId -Comment 'Turned off Store app lookup.'
if (-not $r.Success) { throw "Check-in failed: $($r.Error)" }
```

Publish an approved policy to Active Directory

Publishing writes the approved release of a Universal Policy to its GPO — the same result as **Export to Active Directory** in the console. The policy must be **Approved**.

The **quickest way** is `Deploy-UniversalPolicy`. Pass the policy object and the name of its GPO:

```
$up = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId # must be Approved
Deploy-UniversalPolicy -ThePolicy $up -GPOName 'CORP-Workstation-Hardening' -SkipLinking
```

`-SkipLinking` publishes the settings and leaves the GPO's links as they are. Pass the policy with `-ThePolicy`, as shown, and always name the GPO with `-GPOName`. If you give a GPO name that doesn't exist yet, UPA creates that GPO and publishes to it.

You can also publish by assigning the policy to its GPO with `Assign-UniversalPolicy`, which is the call the console makes. It is the method shown below for a brand-new policy.

To a new GPO. A policy created with `New-UniversalPolicy` has no GPO yet. Leave `AssigneeId` empty and give the name for the new GPO in `AssigneeName`. UPA creates the GPO in the policy's domain, writes the approved settings to it, and records it as the policy's GPO:

```
$up = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId # must be Approved

Assign-UniversalPolicy -UniversalPolicyId $up.GlobalPolicyId -PolicyId ([guid]"$(@($up.Policies)[0].PolicyId)" `
    -AssigneeType GPO -AssigneeId '' -AssigneeName 'CORP-Printer-Baseline' -LinkOnDeployment $false -Activated $true
```

The new GPO is not linked to any OU. Link it where it should apply, in the console or with the link cmdlets in the reference. After the first publish, the policy shows as **CheckedIn** again. That's expected: the release is already published, so don't approve it a second time. From then on, publish each new approved release as below.

To the policy's existing GPO. This covers a policy imported from a GPO, or one you have already published once. The first call creates the assignment. After that, pass the existing `AssignmentId` to publish each new approved release:

```
$up = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId # must be Approved

$assign = @{
    UniversalPolicyId = $up.GlobalPolicyId
    PolicyId          = [guid]"$(@($up.Policies)[0].PolicyId)"
    AssigneeType      = 'GPO'
    AssigneeId        = "$($up.NativePolicyId)" # the GPO's ID
    AssigneeName      = ($up.NativePolicyName -split '::')[-1] # the GPO's name
    LinkOnDeployment  = $false
    Activated         = $true
}

$existing = @(Get-PolicyAssignment -GlobalPolicyId $up.GlobalPolicyId)
if ($existing.Count) { $assign.AssignmentId = $existing[0].AssignmentId }

Assign-UniversalPolicy @assign
```

The GPO normally updates within a few seconds. Confirm the result:

```
Update-UPGPOSyncStatus -UPId $up.GlobalPolicyId      # InSync : True
```

GPOs created outside UPA. UPA writes to a GPO using the domain account it manages that domain with. For a GPO that was created in the Group Policy Management Console and then imported, grant that account **Edit settings, delete, modify security** on the GPO **before** you import it. GPOs that UPA creates already have this permission.

Create and remove a Universal Policy

```
$new = New-UniversalPolicy -UPName 'CORP-Printer-Baseline' -Domain 'contoso.com' -Checkout  
$new.Success
```

`New-UniversalPolicy` needs an account that holds **Create UP**. To remove a policy:

```
Remove-UniversalPolicy -UPId $up.GlobalPolicyId -Force
```

Removing a Universal Policy permanently deletes it and its history from UPA.

Review delegation

```
Get-DelegationAssignments |  
  Select-Object Name,  
    @{ n = 'Role'; e = { $_.Role.Name } },  
    @{ n = 'Users'; e = { ($_.Users | ForEach-Object DisplayName) -join ', ' } },  
    @{ n = 'Views'; e = { ($_.ViewScopes | ForEach-Object Name) -join ', ' } }
```

`Get-Roles` and `Get-ViewScopes` list the Roles and Views on their own.

Read the audit trail

Every change is recorded with the account that made it and the comment it left. The events for one policy, newest first, with each account's name resolved:

```
Get-AuditEvent |  
  Where-Object { $_.GlobalPolicyID -eq $up.GlobalPolicyID } |  
  Sort-Object TimeStamp -Descending |  
  Select-Object TimeStamp, EventType,  
    @{ n = 'User'; e = {  
      ([Security.Principal.SecurityIdentifier]$_ .UserSID).Translate([Security.Principal.NTAccount]).Value } },  
    Comment
```

Add | `Export-Csv -Path .\audit.csv -NoTypeInformation` to keep a copy for an auditor.

Troubleshooting

"The term 'Add-PSSnapin' is not recognized" — you are in PowerShell 7. Open **Windows PowerShell** (`powershell.exe`) instead; `$PSVersionTable.PSVersion` should report 5.1.

"No snap-ins have been registered" — the tools aren't installed on this machine. Install **UPA PowerShell Tools.msi**, then open a new PowerShell window.

"Value does not fall within the expected range" — a general validation message. The real reason is in the error's identifier:

```
$Error[0].FullyQualifiedErrorId
```

The most common is **You must have logged into HAPI to call this cmdlet** — run `Get-HAPIConnection` at the top level of your session, as described in Step 4. Other identifiers name the parameter that needs a value.

A certificate or SSL error when connecting — install a trusted certificate on the UPA server, or, in a test environment only, add `-SkipSslCertificateVerification`.

An empty result, or access denied — your UPA Roles and Views decide what you can see and do. Run `Get-RolesForUser`, and see the **Delegation Guide** for how access is granted.

A command completes but nothing changes — check the returned object's `Success` and `Error` properties, as described under **Move a change through approval**.

The installer won't start — the file is still marked as downloaded. Unblock it, as in Step 1.

Cmdlet reference

The cmdlets are grouped by what they work on. Each entry gives the syntax - one block for each way the cmdlet can be called - followed by its parameters and what it returns. Parameters in square brackets are optional. "In some forms" means a parameter is required in one form of the cmdlet but not in another.

Most cmdlets that act on a Universal Policy accept either `UPId` (the policy's `GlobalPolicyId`) or `ThePolicy` (the policy object returned by `Get-UniversalPolicy`). `Get-Help` works for every cmdlet once the snap-in is loaded, for example `Get-Help Checkin-UniversalPolicy -Full`.

Group	Cmdlets
Connecting and health	4
Universal Policies	15
Change control	7
Reports and sync status	5
Assignments and links	11
OUs, Delegation OUs and the repository	15
Delegation: Roles, Views and Assignments	12
Notifications and audit	5

Group	Cmdlets
Group Policy Objects in Active Directory	18
Active Directory	19
WMI filters	9
Administrative templates	4
Migration maps	3
IP Security	10
Software installation	6
Lookups for policy settings	8
Sign-in and identity providers	29
Agents	2

Connecting and health

Open a session with your UPA server and confirm it is responding.

`Get-HAPIConnection` · `Test-GatekeeperHealth` · `Test-GatewayHealth` · `Get-RolesForUser`

Get-HAPIConnection

Connects to your UPA server.

The `Get-HAPIConnection` cmdlet signs in to your UPA server and opens the session that every other cmdlet uses. Pass a credential (as pipeline input or with `HAPICredentials`) together with `HAPIServerUrl`, or use `SAMLProvider` to sign in through a SAML identity provider. Run it at the top level of your session or script: a connection opened inside a function ends when the function returns.

```
Get-HAPIConnection -HAPICredentials <PSCredential> [-AA] [-UserName <String>]
    [-HAPIServerUrl <String>] [-OutputToken] [-SkipSslCertificateVerification]

Get-HAPIConnection -SAMLProvider <String> [-HAPIServerUrl <String>] [-OutputToken]
    [-SkipSslCertificateVerification]
```

Parameter	Type	Required	Description
HAPICredentials	PSCredential	Yes	A PSCredential containing the DOMAIN\Username and Password to use when connecting to the server. Accepts pipeline input. Alias: cred.
AA	Switch	No	-
UserName	String	No	-
HAPIServerUrl	String	No	The URL of the UPA Gatekeeper. Alias: url.
OutputToken	Switch	No	Switch that will cause the cmdlet to output the retrieved UPA token.

Parameter	Type	Required	Description
SkipSslCertificateVerification	Switch	No	Whether to skip SSL certificate verification. This option should only be used for testing.
SAMLProvider	String	Yes	Performs a SAML login. Alias: saml.

Example

```
$cred = Get-Credential -Message 'UPA account, e.g. admin@contoso.com'
$cred | Get-HAPIConnection -HAPIServerUrl 'https://upa.contoso.com'
```

Test-GatekeeperHealth

Tests to see if the UPA Gatekeeper is responding to calls.

The Test-GatekeeperHealth cmdlet tests to see if the UPA Gatekeeper is responding to calls.

```
Test-GatekeeperHealth -HAPIServerUrl <String>
```

Parameter	Type	Required	Description
HAPIServerUrl	String	Yes	The URL of the UPA Gatekeeper. Alias: url.

Returns: String

Example

```
Test-GatekeeperHealth -HAPIServerUrl 'https://upa.contoso.com'
```

Test-GatewayHealth

Tests to see if the UPA Gateway is responding to calls.

The Test-GatewayHealth cmdlet tests to see if the UPA Gateway is responding to calls.

```
Test-GatewayHealth -HAPIServerUrl <String>
```

Parameter	Type	Required	Description
HAPIServerUrl	String	Yes	The URL of the UPA Gatekeeper. Alias: url.

Returns: String

Get-RolesForUser

Gets the Roles that the currently logged on user is a member of.

The Get-RolesForUser cmdlet Gets the Roles that the currently logged on user is a member of.

```
Get-RolesForUser
```

This cmdlet takes no parameters.

Returns: GetRolesResponse

Example

```
Get-RolesForUser | Select-Object Name, @{ n = 'Role'; e = { $_.Role.Name } }
```

Universal Policies

Find, create, change and remove Universal Policies and their settings.

```
Get-UniversalPolicy · Find-UniversalPolicy · Get-UnassignedGlobalPolicies · New-UniversalPolicy · Import-UniversalPolicy · Clone-UniversalPolicy · Merge-UniversalPolicies · Update-UniversalPolicy · Remove-UniversalPolicy · Get-GPSettings · Find-UniversalPolicySettings · Add-GlobalPolicySetting · Update-UPPolicy · Update-UPSection · Update-PolicyPermissions
```

Get-UniversalPolicy

Gets one or more Universal Policies.

The Get-UniversalPolicy cmdlet gets the properties for a specified Universal Policy, or all Universal Policies.

```
Get-UniversalPolicy [-LoadGPOId <String>] [-UPId <String>] [-PolicyId <String>]
[-SectionId <String>] [-AllDetails] [-PreviousVersion <Boolean>]
[-SpecificVersion <Int>] [-BranchName <String>] [-Domain <String>]
[-OriginalGPOId <String>]
```

Parameter	Type	Required	Description
LoadGPOId	String	No	Obsolete. Use New-UniversalPolicy + Import-UniversalPolicy instead. (optional) Specifies the Guid of a GPO to import into a new UP. Alias: load.
UPId	String	No	Id of the UP to retrieve. If not specified, all UPs will be retrieved.
PolicyId	String	No	If specified, Policy within the UP to retrieve.
SectionId	String	No	If specified, Section within the UP to retrieve.
AllDetails	Switch	No	If this flag is set, the results will include the Policies, Sections, and Settings within the UP. Otherwise, only the UP properties will be returned.
PreviousVersion	Boolean	No	True to retrieve a previous version of the UP.
SpecificVersion	Int	No	If specified, Version of the UP to retrieve.
BranchName	String	No	(Optional) The branch where the Universal Policy is retrieved from.
Domain	String	No	(Optional) The name of the domain where the Universal Policy is retrieved from.

Parameter	Type	Required	Description
OriginalGPOId	String	No	-

Returns: GlobalPolicy

Example

```
Get-UniversalPolicy | Select-Object Name, Version, @{ n = 'Status'; e = { $_.Status.CurrentStatus } }
$up = @(Get-UniversalPolicy | Where-Object { $_.Name -eq 'CORP-Workstation-Hardening' })[0]
```

Find-UniversalPolicy

Finds Universal Policies using the given search filters.

The Find-UniversalPolicy cmdlet performs a search to retrieve Universal Policies that match the specified filters.

```
Find-UniversalPolicy -Filters <WireFilterCriteria[]> [-Domain <String>]
[-MeetsAnyFilter] [-MaxItems <Int>]
```

Parameter	Type	Required	Description
Filters	WireFilterCriteria[]	Yes	Returns only results matching the filters. Accepts pipeline input.
Domain	String	No	The DNS name of the domain in which to search for matching Universal Policies.
MeetsAnyFilter	Switch	No	-
MaxItems	Int	No	If specified, the maximum number of matching Universal Policies to return.

Returns: ItemsResponse<GlobalPolicy>

Get-UnassignedGlobalPolicies

Gets unassigned Universal Policies.

The Get-UnassignedGlobalPolicies cmdlet retrieves all unassigned Universal Policies.

```
Get-UnassignedGlobalPolicies
```

This cmdlet takes no parameters.

Returns: List<GlobalPolicy>

New-UniversalPolicy

Creates a new Universal Policy.

The New-UniversalPolicy cmdlet creates a new Universal Policy in the repository.

```
New-UniversalPolicy -ThePolicy <GlobalPolicy> [-Checkout] [-OUIId <Int>]

New-UniversalPolicy -UPName <String> -Domain <String> [-Checkout] [-OUIId <Int>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	Yes	New UniversalPolicy object to create. Accepts pipeline input.
Checkout	Switch	No	Indicates whether the UP should be checked out.
OUIId	Int	No	Id of the OU to add the UP to.
UPName	String	Yes	The name for the new Universal Policy.
Domain	String	Yes	The DNS name of the domain for the new Universal Policy.

Returns: UpdateGPResponse

Example

```
New-UniversalPolicy -UPName 'CORP-Printer-Baseline' -Domain 'contoso.com' -Checkout
```

Import-UniversalPolicy

Imports a Universal Policy.

The Import-UniversalPolicy cmdlet imports settings from a GPO or other source into a Universal Policy.

```
Import-UniversalPolicy [-UPId <String>] [-BranchName <String>] [-UPName <String>]
[-Checkout] [-AntimalwarePolicyId <String>] [-BaselineId <String>] [-GPOId <String>]
[-DSCScript <String>] [-Domain <String>] [-OverrideGP0SysvolPath <String>]
[-OverrideGP0Details <MsftGpo>] [-OverrideTimestamp <DateTime>]
[-OverrideNativePolicyId <Guid>] [-CheckinComment <String>]
```

Parameter	Type	Required	Description
UPId	String	No	Id of Universal Policy to import into. If not specified, a new Universal Policy will be created.
BranchName	String	No	Optional, Branch name for checked out UP.
UPName	String	No	If a new UP is being created, this specifies the name to use for the new UP.
Checkout	Switch	No	True if the imported UP should be checked out.
AntimalwarePolicyId	String	No	Specifies the Id of an SCCM Anti-malware policy to import into the UP.
BaselineId	String	No	The id of the SCCM Configuration Baseline to import into the Universal Policy
GPOId	String	No	Specifies the Id of a GPO to import into the UP.
DSCScript	String	No	Specifies a DSC script to import into the UP.
Domain	String	No	-

Parameter	Type	Required	Description
OverrideGPOSysvolPath	String	No	-
OverrideGPODetails	MsftGpo	No	-
OverrideTimestamp	DateTime	No	-
OverrideNativePolicyId	Guid	No	-
CheckinComment	String	No	-

Returns: UpdateGPResponse

Clone-UniversalPolicy

Clones a Universal Policy.

The Clone-UniversalPolicy cmdlet clones a Universal Policy, with or without version history.

```
Clone-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-WithHistory]
[-Checkout] [-Name <String>] [-Domain <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	(Optional) The Universal Policy to clone. Accepts pipeline input. Alias: gp.
UPId	Guid	No	(Optional) The Id of the Universal Policy to roll back. Alias: id.
WithHistory	Switch	No	A bool indicating if the version history should be preserved or not. Alias: hist.
Checkout	Switch	No	A bool indicating if the new Universal Policy clone should be left checked out or not. Alias: check.
Name	String	No	(Optional) Name for the newly cloned Universal Policy.
Domain	String	No	(Optional) Name of the domain for the newly cloned Universal Policy.

Returns: UpdateGPResponse

Merge-UniversalPolicies

Merges two Universal Policies into a new Universal Policy.

The Merge-UniversalPolicies cmdlet merges two Universal Policies into a new Universal Policy.

```
Merge-UniversalPolicies -NewUPName <String> [-OU <GPRepositoryOU>] [-OUId <Int>]
[-FirstUP <GlobalPolicy>] [-FirstUPId <Guid>] [-FirstPolicyId <Guid>]
[-SecondUP <GlobalPolicy>] [-SecondUPId <Guid>] [-SecondPolicyId <Guid>]
[-AuthoritativeUPId <Guid>] [-DeleteOriginals <Boolean>]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.

Parameter	Type	Required	Description
OUIId	Int	No	Alias: id.
NewUPName	String	Yes	Alias: name.
FirstUP	GlobalPolicy	No	Alias: up1.
FirstUPId	Guid	No	Alias: upid1.
FirstPolicyId	Guid	No	Alias: pid1.
SecondUP	GlobalPolicy	No	Alias: up2.
SecondUPId	Guid	No	Alias: upid2.
SecondPolicyId	Guid	No	Alias: pid2.
AuthoritativeUPId	Guid	No	Alias: aid.
DeleteOriginals	Boolean	No	Alias: del.

Returns: GlobalPolicy

Update-UniversalPolicy

Updates the properties of a Universal Policy.

The Update-UniversalPolicy cmdlet updates the properties of a Universal Policy. The specified Universal Policy must be checked out before calling this cmdlet.

```
Update-UniversalPolicy -ThePolicy <GlobalPolicy> [-BranchName <String>] [-PropsOnly]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	Yes	Accepts pipeline input.
BranchName	String	No	-
PropsOnly	Switch	No	-

Returns: UpdateGPResponse

Remove-UniversalPolicy

Removes a Universal Policy.

The Remove-UniversalPolicy cmdlet deletes a Universal Policy.

```
Remove-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-Save] [-Force]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: gp.
UPId	Guid	No	Alias: id.

Parameter	Type	Required	Description
Save	Switch	No	-
Force	Switch	No	-

Returns: GeneralSuccessGPResponse

Example

```
Remove-UniversalPolicy -UPId $up.GlobalPolicyId -Force
```

Get-GPSettings

Gets the policy settings for a Universal Policy.

The Get-GPSettings cmdlet gets the policy settings that are defined in the specified Universal Policy.

```
Get-GPSettings [-GPId <String>] [-PolicyId <String>] [-SectionId <String>]
[-PreviousVersion <Boolean>] [-SpecificVersion <Int>]
```

Parameter	Type	Required	Description
GPId	String	No	ID of the Global Policy to retrieve settings for.
PolicyId	String	No	Optional, Id of Policy within the GP to retrieve settings for.
SectionId	String	No	Optional. Id of Section within the GP to retrieve settings for.
PreviousVersion	Boolean	No	Set to True if retrieving a previous version of the GP.
SpecificVersion	Int	No	Optional, Version number of the GP to retrieve (if not set, latest version is retrieved.)

Returns: IDictionary<String, String>[]

Find-UniversalPolicySettings

Finds settings in a given Universal Policy.

The Find-UniversalPolicySettings cmdlet performs a search to retrieve Settings within a given Universal Policies that match the specified filters.

```
Find-UniversalPolicySettings -Filters <WireFilterCriteria[]> [-ThePolicy <GlobalPolicy>]
[-UPId <Guid>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	(Optional) The Universal Policy to search for settings in. Accepts pipeline input. Alias: gp.
UPId	Guid	No	(Optional) The Id of the Universal Policy to search for settings in. Alias: id.
Filters	WireFilterCriteria[]	Yes	The filters used to search for the desired Universal Policy settings.

Returns: List<GPSearchSetting>

Add-GlobalPolicySetting

Adds a policy setting to a Universal Policy.

The Add-GlobalPolicySetting cmdlet adds a policy setting to a Universal Policy object that you have retrieved with Get-UniversalPolicy -AllDetails. The change is made to the object in your session: save it with Update-UniversalPolicy -ThePolicy, then check the policy in. The policy must be checked out. Use SectionId to choose Computer Configuration (the Registry section whose MachinePolicy is True) or User Configuration; the task Move a change through approval shows how, including for a new policy that has no sections yet.

```
Add-GlobalPolicySetting -GP <GlobalPolicy> -Provider <GPPolicyProviderType>
    -Key <String> -ValueName <String> -Value <Object> -ValueType <RegistryValueKind>
    [-PolicyId <Guid>] [-SectionId <Guid>]
```

Parameter	Type	Required	Description
GP	GlobalPolicy	Yes	Accepts pipeline input.
Provider	GPPolicyProviderType	Yes	Values: see Appendix B, GPPolicyProviderType.
PolicyId	Guid	No	-
SectionId	Guid	No	-
Key	String	Yes	-
ValueName	String	Yes	-
Value	Object	Yes	-
ValueType	RegistryValueKind	Yes	Values: String, ExpandString, Binary, DWord, MultiString, QWord, Unknown, None

Returns: GlobalPolicy

Example

```
$u = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId
$pol = @($u.Policies)[0]
$sec = $pol.Sections | Where-Object { $_.PolicyProviderType -eq 'Registry' -and $_.MachinePolicy } | Select-Object
-First 1
Add-GlobalPolicySetting -GP $u -Provider Registry -PolicyId $pol.PolicyId -SectionId $sec.SectionId `
    -Key 'Software\Policies\Microsoft\Windows\Explorer' -ValueName 'NoUseStoreOpenWith' -Value 1 -ValueType DWord |
Out-Null
Update-UniversalPolicy -ThePolicy $u | Out-Null
```

Update-UPPolicy

Updates a Policy within a Universal Policy.

The Update-UPPolicy cmdlet updates the specified Policy within a Universal Policy. The specified Universal Policy must be checked out before calling this cmdlet.

```
Update-UPPolicy -Policy <Policy> -UPId <Guid> [-BranchName <String>] [-PropsOnly]
```

Parameter	Type	Required	Description
Policy	Policy	Yes	Accepts pipeline input.
UPId	Guid	Yes	-
BranchName	String	No	-
PropsOnly	Switch	No	-

Returns: UpdateGPResponse

Update-UPSection

Updates a Policy Section within a Universal Policy.

The Update-UPSection cmdlet updates the specified Policy Section within a Universal Policy. The specified Universal Policy must be checked out before calling this cmdlet.

```
Update-UPSection -Section <PolicySettingProviderBase> -UPId <Guid> -PolicyId <Guid>  
[-BranchName <String>] [-PropsOnly]
```

Parameter	Type	Required	Description
Section	PolicySettingProviderBase	Yes	Accepts pipeline input.
UPId	Guid	Yes	-
PolicyId	Guid	Yes	-
BranchName	String	No	-
PropsOnly	Switch	No	-

Returns: UpdateGPResponse

Update-PolicyPermissions

Updates the permissions on a Universal Policy.

The Update-PolicyPermissions cmdlet sets the access control list (ACL) of a Universal Policy. Use Comment to record the reason for the change.

```
Update-PolicyPermissions -ACL <HapiACL> [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]  
[-PolicyId <String>] [-SectionId <String>] [-BranchName <String>]  
[-Comment <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: up.

Parameter	Type	Required	Description
UPId	Guid	No	Alias: id.
PolicyId	String	No	-
SectionId	String	No	-
BranchName	String	No	-
Comment	String	No	-
ACL	HapiACL	Yes	-

Returns: GeneralSuccessGPRResponse

Change control

Move a Universal Policy through check-out, check-in, approval and release.

[Checkout-UniversalPolicy](#) · [Checkin-UniversalPolicy](#) · [Revert-PolicyCheckout](#) · [Submit-UniversalPolicy](#) · [Approve-UniversalPolicy](#) · [Rollback-UniversalPolicy](#) · [Deploy-UniversalPolicy](#)

Checkout-UniversalPolicy

Checks out a Universal Policy.

The Checkout-UniversalPolicy cmdlet checks out a Universal Policy from the repository, so that it can be modified.

```
Checkout-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-PolicyId <String>]
[-SectionId <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	(Optional) The Universal Policy to check out. Accepts pipeline input. Alias: gp.
UPId	Guid	No	(Optional) The Id of the Universal Policy to check out. Alias: id.
PolicyId	String	No	(optional) PolicyId of a particular policy within the UP. If specified, only the specified Policy is checked out.
SectionId	String	No	(optional) SectionId of a particular policy section within the UP. If specified, only the specified Section is checked out.

Returns: GlobalPolicy

Example

```
Checkout-UniversalPolicy -UPId $up.GlobalPolicyId
```

Checkin-UniversalPolicy

Checks in changes to a Universal Policy.

The Checkin-UniversalPolicy cmdlet checks in changes to a Universal Policy in the repository.

```
Checkin-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]
                        [-BranchName <String>] [-Comment <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Universal Policy object containing the updated settings to check in. Accepts pipeline input. Alias: gp.
UPId	Guid	No	The Universal Policy Id of the policy to check in. Alias: id.
BranchName	String	No	(Optional) The branch where the Universal Policy is currently checked out.
Comment	String	No	(Optional) Comment describing the check in.

Returns: CheckingGPResponse

Example

```
$r = Checkin-UniversalPolicy -UPId $up.GlobalPolicyId -Comment 'Turned off Store app lookup. CR-2041.'
if (-not $r.Success) { throw $r.Error }
```

Revert-PolicyCheckout

Reverts the Checkout of a Universal Policy.

The Revert-PolicyCheckout cmdlet reverts the Checkout of a Universal Policy. Reverting a Checkout moves a Universal Policy from Checked Out to Checked In, without saving any changes that were made to the Universal Policy while it was checked out.

```
Revert-PolicyCheckout [-BranchName <String>] [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]
                    [-PolicyId <String>] [-SectionId <String>] [-Force]
```

Parameter	Type	Required	Description
BranchName	String	No	-
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: up.
UPId	Guid	No	Alias: id.
PolicyId	String	No	-
SectionId	String	No	-
Force	Switch	No	-

Returns: GlobalPolicy

Example

```
Revert-PolicyCheckout -UPId $up.GlobalPolicyId
```

Submit-UniversalPolicy

Submits a checked-in Universal Policy for approval.

The Submit-UniversalPolicy cmdlet submits a checked-in Universal Policy for approval. Its status changes to WaitingForApproval, shown as Pending Approval in the console. A comment is required and is recorded in the policy's history.

```
Submit-UniversalPolicy -Comment <String> [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]  
[-OverrideTimestamp <DateTime>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: gp.
UPId	Guid	No	Alias: id.
Comment	String	Yes	-
OverrideTimestamp	DateTime	No	-

Returns: GlobalPolicy

Example

```
Submit-UniversalPolicy -UPId $up.GlobalPolicyId -Comment 'Ready for review - CR-2041.'
```

Approve-UniversalPolicy

Approves or Unapproves a Universal Policy.

The Approve-UniversalPolicy cmdlet approves or unapproves a Universal Policy.

```
Approve-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-Branch <String>]  
[-Approved <Boolean>] [-Comment <String>] [-OverrideTimestamp <DateTime>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	UniversalPolicyId for the Universal Policy to approve. Accepts pipeline input. Alias: gp.
UPId	Guid	No	(Optional) The Id of the Universal Policy to approve. Alias: id.
Branch	String	No	Branch name to approve (optional).
Approved	Boolean	No	Indicates whether to Approve or Unapprove. Default=true for Approve. Set this flag to false to Unapprove.
Comment	String	No	Optional, comment describing the approval.
OverrideTimestamp	DateTime	No	-

Returns: CheckinGPResponse

Example

```
Approve-UniversalPolicy -UPId $up.GlobalPolicyId -Approved $true -Comment 'Reviewed against CR-2041.'
```

Rollback-UniversalPolicy

Rolls back a Universal Policy to a previous version.

The Rollback-UniversalPolicy cmdlet rolls back a Universal Policy to a previous version.

```
Rollback-UniversalPolicy -RollbackVersion <Int> [-ThePolicy <GlobalPolicy>]  
[-UPId <Guid>] [-Comment <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: up.
UPId	Guid	No	Alias: id.
RollbackVersion	Int	Yes	Alias: ver.
Comment	String	No	-

Returns: GeneralSuccessGPResponse

Deploy-UniversalPolicy

Deploys a Universal Policy.

The Deploy-UniversalPolicy cmdlet publishes the approved release of a Universal Policy to a GPO in Active Directory and returns the Id of the deployment job. Pass the policy object with ThePolicy and name the target GPO with GPOName; if no GPO of that name exists, UPA creates it. Use SkipLinking to publish the settings without changing the GPO's links. See the task Publish an approved policy to Active Directory.

```
Deploy-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-PolicyId <String>]  
[-SkipLinking] [-ReleaseVersion <Int>] [-GPOId <String>] [-GPOName <String>]  
[-Domain <String>] [-DeferUntil <DateTime>] [-AntimalwarePolicyId <String>]  
[-AntimalwarePolicyName <String>] [-AntimalwareCollection <String>]  
[-BaselineCollection <String>] [-BaselineId <String>] [-BaselineName <String>]  
[-DSCScript <String>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	Accepts pipeline input. Alias: gp.
UPId	Guid	No	Alias: id.
PolicyId	String	No	-
SkipLinking	Switch	No	Indicates whether the Universal Policy links should not be updated on AD OUs. Alias: nolink.

Parameter	Type	Required	Description
ReleaseVersion	Int	No	Alias: ver.
GPOId	String	No	-
GPOName	String	No	-
Domain	String	No	-
DeferUntil	DateTime	No	-
AntimalwarePolicyId	String	No	-
AntimalwarePolicyName	String	No	-
AntimalwareCollection	String	No	-
BaselineCollection	String	No	-
BaselineId	String	No	-
BaselineName	String	No	-
DSCScript	String	No	-

Returns: String

Example

```
$up = Get-UniversalPolicy -AllDetails -UPId $up.GlobalPolicyId
Deploy-UniversalPolicy -ThePolicy $up -GPOName 'CORP-Workstation-Hardening' -SkipLinking
```

Reports and sync status

Report on a policy's settings and compare it with other versions or with its GPO.

[Get-SettingsReport](#) · [Get-DiffReport](#) · [Get-RSOPReport](#) · [Update-UPGPOSyncStatus](#) · [Update-UPReplicaUPSyncStatus](#)

Get-SettingsReport

Generates a Settings report for a Universal Policy.

The Get-SettingsReport cmdlet generates a Settings report for the specified Universal Policy.

```
Get-SettingsReport -Id <Guid> [-Version <String>] [-Master] [-Release]
```

Parameter	Type	Required	Description
Id	Guid	Yes	-
Version	String	No	-
Master	Switch	No	-

Parameter	Type	Required	Description
Release	Switch	No	-

Returns: The Universal Policy (GlobalPolicy).

Get-DiffReport

Generates a report that compares two versions of a Universal Policy.

The Get-DiffReport cmdlet generates a report that compares two versions of a Universal Policy.

```
Get-DiffReport [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-Versions <Int[]>]
  [-Last2Versions <Boolean>] [-OutFile <String>] [-CheckedOutVersusCheckedIn]
  [-VersusApproved] [-VersusGPOInAD] [-SecondUPId <Guid>]
  [-SecondPolicy <GlobalPolicy>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	The Universal Policy on which to perform the check. Accepts pipeline input. Alias: gp.
UPId	Guid	No	The Id of the Universal Policy whose differences to report.
Versions	Int[]	No	The versions of this Universal Policy to compare.
Last2Versions	Boolean	No	Whether to compare the last two version of this Universal Policy.
OutFile	String	No	The path of the File to write the report to.
CheckedOutVersusCheckedIn	Switch	No	-
VersusApproved	Switch	No	-
VersusGPOInAD	Switch	No	-
SecondUPId	Guid	No	-
SecondPolicy	GlobalPolicy	No	A different Universal Policy whose differences to report against the other given Universal Policy.

Returns: GlobalPolicyDiffReportResults - Changes and Matches, with GlobalPolicyId and GlobalPolicyName.

Example

```
$diff = Get-DiffReport -UPId $up.GlobalPolicyId -VersusGPOInAD
$diff.Changes.Count
```

Get-RSOPReport

Generates an RSOP report for a Universal Policy.

The Get-RSOPReport cmdlet generates an RSOP report for a Universal Policy.

```
Get-RSOPReport -Id <Guid> [-Version <String>]
```

Parameter	Type	Required	Description
Id	Guid	Yes	-
Version	String	No	-

Returns: String

Update-UPGPOSyncStatus

Checks if the given Universal Policy is in sync with the AD GPO it was created from.

The Update-UPGPOSyncStatus cmdlet checks if the given Universal Policy is in sync with the AD GPO it was created from.

```
Update-UPGPOSyncStatus [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	The Universal Policy on which to perform the check. Accepts pipeline input. Alias: gp.
UPId	Guid	No	The Id of the Universal Policy on which to perform the check. Alias: id.

Returns: An object with InSync and ReleaseOutOfSync (Boolean).

Example

```
Update-UPGPOSyncStatus -UPId $up.GlobalPolicyId
```

Update-UPReplicaUPSyncStatus

Updates the sync status between the given Universal Policy and its replica UPs.

The Update-UPReplicaUPSyncStatus cmdlet updates the sync status between the given Universal Policy and its replica UPs.

```
Update-UPReplicaUPSyncStatus [-ThePolicy <GlobalPolicy>] [-UPId <Guid>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	The Universal Policy on which to perform the check. Accepts pipeline input. Alias: gp.
UPId	Guid	No	The Id of the Universal Policy on which to perform the check. Alias: id.

Returns: Boolean

Assignments and links

Assign Universal Policies and link them to OUs and other scopes.

Assign-UniversalPolicy

Assigns a Universal Policy to a GPO, agent or other target. Assigning an approved policy to its GPO publishes it to Active Directory.

The Assign-UniversalPolicy cmdlet assigns a Universal Policy to a target. With AssigneeType GPO it publishes the approved release of the policy to that GPO - the same operation as Export to Active Directory in the console. With an empty AssigneeId and a new name in AssigneeName, it creates a new, unlinked GPO and publishes to it. The first call creates the assignment; to publish a later approved release, call it again with the AssignmentId returned by Get-PolicyAssignment. See the task Publish an approved policy to Active Directory.

```
Assign-UniversalPolicy [-AssignmentId <Int>] [-UniversalPolicyId <Guid>]
  [-PolicyId <Guid>] [-DeploymentTime <DateTime>] [-Activated <Boolean>]
  [-LinkOnDeployment <Boolean>] [-JobIds <String[]>]
  [-AssigneeType <DeploymentAssigneeType>] [-AssigneeId <String>]
  [-AssigneeName <String>]
```

Parameter	Type	Required	Description
AssignmentId	Int	No	The unique Id of the assignment table entry.
UniversalPolicyId	Guid	No	Universal Policy Id holding the assigned policy.
PolicyId	Guid	No	The assigned Policy's Id.
DeploymentTime	DateTime	No	Date and time to deploy this UP.
Activated	Boolean	No	-
LinkOnDeployment	Boolean	No	Boolean indicating if the policy needs to be linked upon deployment.
JobIds	String[]	No	An array of strings holding the job Ids associated with deployments.
AssigneeType	DeploymentAssigneeType	No	The Type of the Assignee. Values: see Appendix B, DeploymentAssigneeType.
AssigneeId	String	No	The Id of the assignee (like the Agent Id).
AssigneeName	String	No	A Name of the assignee.

Returns: GPPolicyAssignment

Example

```
# publish the approved release of $up to its GPO
$a = @(Get-PolicyAssignment -GlobalPolicyId $up.GlobalPolicyId)
Assign-UniversalPolicy -AssignmentId $a[0].AssignmentId -UniversalPolicyId $up.GlobalPolicyId `
  -PolicyId ([guid]"${@($up.Policies)[0].PolicyId}") -AssigneeType GPO `
  -AssigneeId "$($up.NativePolicyId)" -AssigneeName ($up.NativePolicyName -split '::-')[1] `
  -LinkOnDeployment $false -Activated $true
```

Get-PolicyAssignment

Gets the Policy Assignments for a Universal Policy, or Assignee.

The Get-PolicyAssignment cmdlet gets the Policy Assignments for a Universal Policy or Assignee.

```
Get-PolicyAssignment [-AssignmentId <Int>] [-GlobalPolicyId <Guid>] [-PolicyId <Guid>]
[-AssigneeId <String>] [-AssigneeName <String>] [-Activated <Boolean>]
```

Parameter	Type	Required	Description
AssignmentId	Int	No	The optional Id of the policy deployment assignment to get.
GlobalPolicyId	Guid	No	The optional Id of the GlobalPolicy to get policy deployment assignments for.
PolicyId	Guid	No	The optional Id of the Policy to get it's deployment assignments. If you specify this PolicyId, then you must also supply the GlobalPolicyId.
AssigneeId	String	No	The optional Id of an assignee to get all deployment assignments for this assignee.
AssigneeName	String	No	The optional name of an assignee to get all deployment assignments for this assignee.
Activated	Boolean	No	The optional activation status to get all deployment assignments with this status.

Returns: List<GPPolicyAssignment>

Remove-PolicyAssignment

Removes a Policy Assignment.

The Remove-PolicyAssignment cmdlet removes a policy assignment.

```
Remove-PolicyAssignment [-PolicyAssignment <GPPolicyAssignment>] [-AssignmentId <Int>]
```

Parameter	Type	Required	Description
PolicyAssignment	GPPolicyAssignment	No	Accepts pipeline input. Alias: pa.
AssignmentId	Int	No	Alias: aid.

Set-PolicyActivation

Activates or Deactivates a Policy Assignment.

The Set-PolicyActivation cmdlet Activates or Deactivates a Policy Assignment. Policies will only be applied to Agents when the Policy Assignment is Activated.

```
Set-PolicyActivation [-PolicyAssignment <GPPolicyAssignment>] [-AssignmentId <Int>]
[-Activate <Boolean>]
```

Parameter	Type	Required	Description
PolicyAssignment	GPPolicyAssignment	No	Accepts pipeline input. Alias: pa.

Parameter	Type	Required	Description
AssignmentId	Int	No	Alias: aid.
Activate	Boolean	No	-

Lock-Policy

Locks a Universal Policy to a GPO.

The Lock-Policy cmdlet helps insure that the settings in a GPO match the settings in the associated Universal Policy by monitoring the GPO settings, and optionally updating the GPO settings to match the Universal Policy.

```
Lock-Policy [-AssignmentId <Int>] [-LockStatus <PolicyLockStatus>]
```

Parameter	Type	Required	Description
AssignmentId	Int	No	-
LockStatus	PolicyLockStatus	No	Values: None, Monitor, Lock, Remediate

Returns: GPPolicyAssignment

New-PolicyLink

Links a Universal Policy to a Scope.

The New-PolicyLink cmdlet adds a new link, linking a Universal Policy to a Scope.

```
New-PolicyLink [-UPId <Guid>] [-PolicyId <Guid>] [-AssignmentId <Int>]
  [-Assignee <GPPolicyAssignee>] [-LinkSite <LinkSiteInfo>] [-LinkOrder <Int>]
```

Parameter	Type	Required	Description
UPId	Guid	No	-
PolicyId	Guid	No	-
AssignmentId	Int	No	-
Assignee	GPPolicyAssignee	No	-
LinkSite	LinkSiteInfo	No	-
LinkOrder	Int	No	-

Returns: PolicyLinkInfo

Get-PolicyLink

Gets the Policy Links for a Universal Policy, or Assignee.

The Get-PolicyLink cmdlet gets the Policy Links for a Universal Policy or Assignee.

```
Get-PolicyLink [-LinkId <Int>] [-AssgineeUniqueId <Int>] [-GlobalPolicyId <Guid>]
[-PolicyId <Guid>] [-AssigneeId <String>]
```

Parameter	Type	Required	Description
LinkId	Int	No	The Id of a specific link to get.
AssgineeUniqueld	Int	No	The Id of an assignee to get all links for.
GlobalPolicyId	Guid	No	The optional Id of the GlobalPolicy to get policy links for.
PolicyId	Guid	No	The optional Id of the Policy to get it's links. If you specify this PolicyId, then you must also supply the GlobalPolicyId.
Assigneeld	String	No	The string identifying an assignee to get all links for.

Returns: List<PolicyLinkInfo>

Unlink-PolicyFromScope

Unlinks a Policy from a Scope.

The Unlink-PolicyFromScope cmdlet unlinks a policy from a scope container, such as an OU.

```
Unlink-PolicyFromScope [-ContainerTypeId <String>] [-ContainerId <String>]
[-UPId <Guid>] [-PolicyId <Guid>]
```

Parameter	Type	Required	Description
ContainerTypeld	String	No	-
ContainerId	String	No	-
UPId	Guid	No	-
PolicyId	Guid	No	-

Link-UniversalPolicy

Updates all the AD OU Links that the given Universal Policy is linked to.

The Link-UniversalPolicy cmdlet updates all the AD OU Links that the given Universal Policy is linked to.

```
Link-UniversalPolicy [-ThePolicy <GlobalPolicy>] [-UPId <Guid>] [-PolicyId <Guid>]
```

Parameter	Type	Required	Description
ThePolicy	GlobalPolicy	No	The Universal Policy whose AD OU links should be updated. Accepts pipeline input. Alias: gp.
UPId	Guid	No	The Id of the Univeral Policy whose AD OU links should be updated. Alias: id.

Parameter	Type	Required	Description
PolicyId	Guid	No	The Id of the specific policy in the Universal Policy whose AD OU links should be updated. Alias: pid.

Returns: String

Get-LinkSites

Gets the set of objects a Universal Policy can be linked to.

The Get-LinkSites cmdlet gets the set of objects a Universal Policy can be linked to.

```
Get-LinkSites [-PolicyType <GPPolicyType>] [-ContainerType <String>]
```

Parameter	Type	Required	Description
PolicyType	GPPolicyType	No	Values: GPO, DSC, Linux, Mac, Office365, Box, SCCM, Intune, EDR
ContainerType	String	No	-

Returns: ItemsResponse<LinkSiteInfo>

Get-LinkSiteType

Gets the types of scopes a specified policy type can be linked to.

The Get-LinkSiteType cmdlet gets the types of scopes that a specified policy type can be linked to. For example, GPO policies can be linked to Sites or OUs.

```
Get-LinkSiteType [-PolicyType <GPPolicyType>]
```

Parameter	Type	Required	Description
PolicyType	GPPolicyType	No	The type of the Policy whose types of scopes it can be linked to is retrieved. Values: GPO, DSC, Linux, Mac, Office365, Box, SCCM, Intune, EDR

Returns: List<LinkSiteType>

OUs, Delegation OUs and the repository

Work with the OU trees UPA manages, and load Active Directory into the repository.

```
Get-AllOUs · Get-RepositoryOU · Get-CloudOU · Add-CloudOU · Add-DelegationOU · Rename-OU · Remove-OU · Get-OUChild · Add-ChildToOU · Remove-OUChild · Get-OULinks · Update-OULinkOrder · Load-ADRepository · Stop-LoadRepositoryJob · Get-JobProgress
```

Get-AllOUs

Gets Repository and Cloud OUs.

The Get-AllOUs cmdlet retrieves all Repository and Cloud OUs without linked children details.

```
Get-AllOUs
```

This cmdlet takes no parameters.

Returns: The Delegation OUs and the domain root. For the OUs in Active Directory, use Get-RepositoryOU.

Get-RepositoryOU

Gets one or more Repository OUs.

The Get-RepositoryOU cmdlet retrieves one or more Repository OUs.

```
Get-RepositoryOU [-OUIId <Int>] [-OUDN <String>] [-ParentOUIId <Int>]
[-ParentOUDN <String>] [-GlobalPolicyId <String>] [-AgentId <Int>] [-LinkId <Int>]
[-AssigneeId <Int>] [-OnlyRootOUs]
```

Parameter	Type	Required	Description
OUIId	Int	No	(Optional) Id of the specific Repository OU to retrieve. Alias: id.
OUDN	String	No	(Optional) The distinguished name of the specific Repository OU to retrieve. Alias: dn.
ParentOUIId	Int	No	(Optional) The Id of the parent whose children OUs should be retrieved. Alias: pid.
ParentOUDN	String	No	(Optional) The distinguished name of the parent whose children OUs should be retrieved. Alias: pdn.
GlobalPolicyId	String	No	(Optional) The Id of the Global Policy whose linked OUs are to be retrieved. Alias: gpid.
AgentId	Int	No	(Optional) The Id of the agent whose containing OU should be retrieved. Alias: agid.
LinkId	Int	No	(Optional) The Id of the Link whose linked OU should be retrieved. Alias: lid.
AssigneeId	Int	No	(Optional) The Id of the assignee whose linked OUs should be retrieved. Alias: aid.
OnlyRootOUs	Switch	No	(Optional) A switch indicating if only root level OUs (with no parent) are to be retrieved. Alias: root.

Returns: List<GPRepositoryOU>

Example

```
Get-RepositoryOU | Select-Object Name
```

Get-CloudOU

Gets one or more Cloud OUs.

The Get-CloudOU cmdlet retrieves Cloud OUs.

```
Get-CloudOU [-OUIId <Int>] [-OUDN <String>] [-ParentOUIId <Int>] [-ParentOUDN <String>]
[-UPIId <String>] [-AgentId <Int>] [-LinkId <Int>] [-AssigneeId <Int>] [-OnlyRootOUs]
```

Parameter	Type	Required	Description
OUIId	Int	No	(Optional) Id of the specific Cloud OU to retrieve. Alias: id.
OUDN	String	No	(Optional) The distinguished name of the specific Cloud OU to retrieve. Alias: dn.
ParentOUIId	Int	No	(Optional) The Id of the parent whose children OUs should be retrieved. Alias: pid.
ParentOUDN	String	No	(Optional) The distinguished name of the parent whose children OUs should be retrieved. Alias: pdn.
UPIId	String	No	(Optional) The Id of the Universal Policy whose linked OUs are to be retrieved.
AgentId	Int	No	(Optional) The Id of the agent whose containing OU should be retrieved. Alias: agid.
LinkId	Int	No	(Optional) The Id of the Link whose linked OU should be retrieved. Alias: lid.
AssigneeId	Int	No	(Optional) The Id of the assignee whose linked OUs should be retrieved. Alias: aid.
OnlyRootOUs	Switch	No	(Optional) A switch indicating if only root level OUs (with no parent) are to be retrieved. Alias: root.

Returns: ItemsResponse<GPRepositoryOU>

Add-CloudOU

Adds a new Cloud OU into the Cloud OU tree.

The Add-CloudOU cmdlet adds a new Cloud OU into the Cloud OU tree.

```
Add-CloudOU -Name <String> [-ParentOU <GPRepositoryOU>] [-ParentOUIId <Int>]
[-Description <String>]
```

Parameter	Type	Required	Description
ParentOU	GPRepositoryOU	No	Accepts pipeline input. Alias: pou.
ParentOUIId	Int	No	Alias: pid.
Name	String	Yes	-
Description	String	No	Alias: desc.

Returns: GPRepositoryOU

Add-DelegationOU

Adds a new Delegation OU into the Delegation OU tree.

The Add-DelegationOU cmdlet adds a new Delegation OU into the Delegation OU tree.

```
Add-DelegationOU -Name <String> [-ParentOU <GPRepositoryOU>] [-ParentOUIId <Int>]
[-Description <String>]
```

Parameter	Type	Required	Description
ParentOU	GPRepositoryOU	No	Accepts pipeline input. Alias: pou.
ParentOUIId	Int	No	Alias: pid.
Name	String	Yes	-
Description	String	No	Alias: desc.

Returns: GPRepositoryOU

Rename-OU

Renames a Cloud or Delegation OU.

The Rename-OU cmdlet renames a Cloud or Delegation OU.

```
Rename-OU -NewName <String> [-OU <GPRepositoryOU>] [-OUIId <Int>]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.
OUIId	Int	No	Alias: id.
NewName	String	Yes	Alias: name.

Remove-OU

Removes a Cloud or Repository OU.

The Remove-OU cmdlet removes a Cloud or Repository OU.

```
Remove-OU [-OU <GPRepositoryOU>] [-OUIId <Int>] [-JustDescendants]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.
OUIId	Int	No	Alias: id.
JustDescendants	Switch	No	Alias: children.

Get-OUChild

Gets the child objects in a Cloud or Repository OU.

The Get-OUChild cmdlet retrieves one or more Cloud or Repository OU children (Agents or Universal Policies).

```
Get-OUChild [-OU <GPRepositoryOU>] [-OUIId <Int>] [-LinkId <Int>] [-AssigneeId <Int>]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	The OU whose children to retrieve. Accepts pipeline input.
OUIId	Int	No	The Id of the OU whose children to retrieve. Alias: id.
LinkId	Int	No	(Optional) The Id of the Link whose linked OU's children should be retrieved. Alias: lid.
AssigneeId	Int	No	(Optional) The Id of the assignee whose linked OU's children should be retrieved. Alias: aid.

Returns: ItemsResponse<GetOUChildrenResponse>

Add-ChildToOU

Adds agents or Universal Policies to a Cloud or Repository OU.

The Add-ChildToOU cmdlet adds a child or children (agents or UPs) to a Cloud or Repository OU.

```
Add-ChildToOU [-OU <GPRepositoryOU>] [-OUIId <Int>] [-AgentIds <Int[]>]
[-GPs <GPChild[]>]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.
OUIId	Int	No	Alias: id.
AgentIds	Int[]	No	-
GPs	GPChild[]	No	-

Returns: ItemsResponse<PolicyLinkInfo>

Remove-OUChild

Removes one or more children from a Cloud or Repository OU.

The Remove-OUChild cmdlet removes one or more children (Agents or Universal Policies) from a Cloud or Repository OU.

```
Remove-OUChild [-OU <GPRepositoryOU>] [-OUIId <Int>] [-AgentIds <Int[]>]
[-UPIIds <Guid[]>] [-RemoveAllChildren]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.
OUIId	Int	No	Alias: id.
AgentIds	Int[]	No	-

Parameter	Type	Required	Description
UPIds	Guid[]	No	-
RemoveAllChildren	Switch	No	Alias: all.

Get-OULinks

Gets the Universal Policies and/or Agents that are linked to a Cloud or Repository OU.

The Get-OULinks cmdlet retrieves the Universal Policies and/or Agent links to a Cloud or Repository OU.

```
Get-OULinks [-OU <GPRepositoryOU>] [-OUIId <Int>] [-LinkId <Int>] [-GPLinksOnly]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	The OU whose links to retrieve. Accepts pipeline input.
OUIId	Int	No	The Id of the OU whose links to retrieve. Alias: id.
LinkId	Int	No	(Optional) The Id of the OU Link to retrieve. Alias: lid.
GPLinksOnly	Switch	No	A switch indicating if only Global Policy links should be retrieved. Alias: gp.

Returns: List<PolicyLinkInfo>

Update-OULinkOrder

Updates the link order for a child Universal Policy to a Cloud or Repository OU.

The Update-OULinkOrder cmdlet updates the link order for a child Universal Policy to a Cloud or Repository OU. .

```
Update-OULinkOrder [-OU <GPRepositoryOU>] [-OUIId <Int>] [-LinkId <Int>]
[-NewPosition <Int>]
```

Parameter	Type	Required	Description
OU	GPRepositoryOU	No	Accepts pipeline input.
OUIId	Int	No	Alias: id.
LinkId	Int	No	Alias: lid.
NewPosition	Int	No	Alias: pos.

Load-ADRepository

Uploads OU and GPO information from a domain into the repository.

The Load-ADRepository cmdlet loads information from a domain into the repository. Information uploaded includes OU tree structure, and Group Policy settings.

```
Load-ADRepository [-Domain <String>] [-TargetedContainer <String>]
```

Parameter	Type	Required	Description
Domain	String	No	The name of the domain to load into the Repository.
TargetedContainer	String	No	The AD container\OU to limit our import to. Alias: targ.

Returns: String

Stop-LoadRepositoryJob

Stops a job that is loading AD OUs and GPOs into the repository.

The Stop-LoadRepositoryJob cmdlet stops a job that is loading AD OUs and GPOs into the repository. The job could have been started using the Import action on the Repository node in the UI or the job started by the Load-ADRepository cmdlet.

```
Stop-LoadRepositoryJob [-Comment <String>]
```

Parameter	Type	Required	Description
Comment	String	No	(Optional) Comment describing the reason for stopping the Load Repository job.

Returns: String

Get-JobProgress

Retrieves the completion percentage for the given job.

The Get-JobProgress cmdlet retrieves the completion percentage for the given job (i.e. a Load Repository or Export to AD job). The job Id is returned from the operation that starts such a job.

```
Get-JobProgress -JobId <Guid>
```

Parameter	Type	Required	Description
JobId	Guid	Yes	Accepts pipeline input. Alias: id.

Delegation: Roles, Views and Assignments

Manage who can do what, and where. The Delegation Guide explains how these fit together.

[Get-Roles](#) ·
 [Add-Role](#) ·
 [Update-Role](#) ·
 [Remove-Role](#) ·
 [Get-ViewScopes](#) ·
 [Add-ViewScope](#) ·
 [Update-ViewScope](#) ·
 [Remove-ViewScope](#) ·
 [Get-DelegationAssignments](#) ·
 [Add-DelegationAssignment](#) ·
 [Update-DelegationAssignment](#) ·
 [Remove-DelegationAssignment](#)

Get-Roles

Gets the list of Roles.

The Get-Roles cmdlet gets the list of roles. Roles define what someone can do to Global Policy for delegation purposes.

```
Get-Roles [-Name <String>] [-RoleId <Int>]
```

Parameter	Type	Required	Description
Name	String	No	Name of a specific role to retrieve.
RoleId	Int	No	Id of a specific role to retrieve. Alias: id.

Returns: HAPIRole2[]

Add-Role

Creates a new Role.

This cmdlet creates a new Role. Roles define what someone can do to Global Policy for delegation purposes.

```
Add-Role -Name <String> -Permissions <Permissions>
```

Parameter	Type	Required	Description
Name	String	Yes	Name for the new role.
Permissions	Permissions	Yes	Permissions for the new role. Values: see Appendix B, Permissions. Alias: perms.

Returns: HAPIRole2

Update-Role

Updates a role's name or permissions.

This cmdlet updates a role. Roles define what someone can do to Global Policy for delegation purposes.

```
Update-Role [-Role <HAPIRole2>] [-Name <String>] [-Permissions <Permissions>]  
[-RoleId <Int>]
```

Parameter	Type	Required	Description
Role	HAPIRole2	No	A role object with properties to be updated. Accepts pipeline input.
Name	String	No	New name for the role.
Permissions	Permissions	No	New permissions for the role. Values: see Appendix B, Permissions. Alias: perms.
RoleId	Int	No	The Id of the role be updated. Alias: id.

Returns: HAPIRole2

Remove-Role

Removes a role.

The Remove-Role cmdlet removes a role. Roles define what someone can do to Global Policy for delegation purposes.

```
Remove-Role [-Role <HAPIRole2>] [-RoleId <Int>]
```

Parameter	Type	Required	Description
Role	HAPIRole2	No	A role object to be removed. Accepts pipeline input.
RoleId	Int	No	The Id of the role to be removed. Alias: id.

Returns: DirectoryGeneralResponse

Get-ViewScopes

Gets the list of View Scopes.

The Get-ViewScopes cmdlet gets the list of view scopes. View scopes define what someone can see for Global Policy delegation purposes.

```
Get-ViewScopes [-Name <String>] [-ViewScopeId <Int>]
```

Parameter	Type	Required	Description
Name	String	No	Name of a specific view scope to retrieve.
ViewScopeId	Int	No	Id of a specific view scope to retrieve. Alias: id.

Returns: ViewScope[]

Add-ViewScope

Creates a new View Scope.

This cmdlet creates a new View Scope. View scopes define what someone can see for Global Policy delegation purposes.

```
Add-ViewScope -Rules <ViewRule[]> -Name <String>
```

Parameter	Type	Required	Description
Rules	ViewRule[]	Yes	Rules for the new view scope. Accepts pipeline input.
Name	String	Yes	Name for the new view scope.

Returns: ViewScope

Update-ViewScope

Updates a view scope's name or rules.

This cmdlet updates a view scope. View scopes define what someone can see for Global Policy delegation purposes.

```
Update-ViewScope [-ViewScope <ViewScope>] [-Name <String>] [-Rules <ViewRule[]>]  
[-ViewScopeId <Int>]
```

Parameter	Type	Required	Description
ViewScope	ViewScope	No	A view scope object with properties to be updated. Accepts pipeline input.
Name	String	No	New name for the view scope.
Rules	ViewRule[]	No	The new rules for the view scope.
ViewScopeId	Int	No	The Id of the view scope be updated. Alias: id.

Returns: ViewScope

Remove-ViewScope

Removes a View Scope.

The Remove-ViewScope cmdlet removes a view scope. View scopes define what someone can see for Global Policy delegation purposes.

```
Remove-ViewScope [-ViewScope <ViewScope>] [-ViewScopeId <Int>]
```

Parameter	Type	Required	Description
ViewScope	ViewScope	No	A view scope object to be removed. Accepts pipeline input.
ViewScopeId	Int	No	The Id of the view scope to be removed. Alias: id.

Returns: DirectoryGeneralResponse

Get-DelegationAssignments

Gets the list of delegation assignments.

The Get-DelegationAssignments cmdlet gets the list of delegation assignments. Delegation assignments define who can see what and do what for Global Policy delegation purposes.

```
Get-DelegationAssignments [-Name <String>] [-DelegationAssignmentId <Int>]
  [-ViewScopeId <Int>] [-RoleId <Int>] [-UserId <Long>] [-GroupId <Long>]
  [-UserSID <String>] [-GroupSID <String>]
```

Parameter	Type	Required	Description
Name	String	No	Name of a specific delegation assignment to retrieve.
DelegationAssignmentId	Int	No	Id of a specific delegation assignment to retrieve. Alias: id.
ViewScopeId	Int	No	Id of a view scope whose containing delegation assignments to retrieve. Alias: vsid.
RoleId	Int	No	Id of a role whose containing delegation assignments to retrieve. Alias: rid.
UserId	Long	No	Id of a user whose assigned delegation assignments to retrieve. Alias: uid.
GroupId	Long	No	Id of a group whose assigned delegation assignments to retrieve. Alias: gid.

Parameter	Type	Required	Description
UserSID	String	No	SID of a user whose assigned delegation assignments to retrieve. Alias: usid.
GroupSID	String	No	SID of a group whose assigned delegation assignments to retrieve. Alias: gsid.

Returns: DelegationVRAssignment[]

Example

```
Get-DelegationAssignments |
  Select-Object Name, @{ n = 'Users'; e = { ($_.Users | ForEach-Object DisplayName) -join ', ' } }
```

Add-DelegationAssignment

Creates a new delegation assignment.

This cmdlet creates a new delegation assignment. Delegation assignments define who can see what and do what for Global Policy delegation purposes.

```
Add-DelegationAssignment -Name <String> [-ViewScopes <ViewScope[]>]
  [-ViewScopeIds <Int[]>] [-Description <String>] [-Role <HAPIRole2>] [-RoleId <Int>]
  [-Users <ADUserEntry[]>] [-UserSIDs <String[]>] [-Groups <ADGroupEntry[]>]
  [-GroupSIDs <String[]>]
```

Parameter	Type	Required	Description
ViewScopes	ViewScope[]	No	View scopes to be part of this new delegation assignment. Accepts pipeline input. Alias: vss.
ViewScopeIds	Int[]	No	Ids of view scopes to be part of this new delegation assignment. Alias: vsids.
Name	String	Yes	Name for the new delegation assignment.
Description	String	No	Description for the new delegation assignment.
Role	HAPIRole2	No	The Role to be part of this new delegation assignment.
RoleId	Int	No	Id of the role to be part of this new delegation assignment. Alias: rid.
Users	ADUserEntry[]	No	Users to be part of this new delegation assignment.
UserSIDs	String[]	No	SIDs of users to be part of this new delegation assignment. Alias: usid.
Groups	ADGroupEntry[]	No	Groups to be part of this new delegation assignment.
GroupSIDs	String[]	No	SIDs of groups to be part of this new delegation assignment. Alias: gsid.

Returns: DelegationVRAssignment

Update-DelegationAssignment

Updates a delegation assignment's name, description, or contained role, view scopes, users or groups.

This cmdlet updates a delegation assignment. Delegation assignments define who can see what and do what for Global Policy delegation purposes.

```
Update-DelegationAssignment [-DelegationAssignment <DelegationVRAssignment>]
[-DelegationAssignmentId <Int>] [-Name <String>] [-Description <String>]
[-ViewScopes <ViewScope[]>] [-ViewScopeIds <Int[]>] [-Role <HAPIRole2>]
[-RoleId <Int>] [-Users <ADUserEntry[]>] [-UserSIDs <String[]>]
[-Groups <ADGroupEntry[]>] [-GroupSIDs <String[]>]
```

Parameter	Type	Required	Description
DelegationAssignment	DelegationVRAssignment	No	A delegation assignment object with properties to be updated. Accepts pipeline input.
DelegationAssignmentId	Int	No	The Id of the delegation assignment be updated. Alias: id.
Name	String	No	The new name for the delegation assignment.
Description	String	No	The new description for the delegation assignment.
ViewScopes	ViewScope[]	No	The new view scopes for the delegation assignment. Alias: vss.
ViewScopeIds	Int[]	No	Ids of the new view scopes for the delegation assignment. Alias: vsids.
Role	HAPIRole2	No	The new role for the delegation assignment.
RoleId	Int	No	Id of the new role for the delegation assignment. Alias: rid.
Users	ADUserEntry[]	No	New users for the delegation assignment.
UserSIDs	String[]	No	SIDs of new users for the delegation assignment. Alias: usid.
Groups	ADGroupEntry[]	No	New groups for the delegation assignment.
GroupSIDs	String[]	No	SIDs of new groups for the delegation assignment. Alias: gsid.

Returns: DelegationVRAssignment

Remove-DelegationAssignment

Removes a delegation assignment.

The Remove-DelegationAssignmentcmdlet removes a delegation assignment. Delegation assignments define who can see what and do what for Global Policy delegation purposes.

```
Remove-DelegationAssignment [-DelegationAssignment <DelegationVRAssignment>]
[-DelegationAssignmentId <Int>]
```

Parameter	Type	Required	Description
DelegationAssignment	DelegationVRAssignment	No	A delegation assignment object to be removed. Accepts pipeline input. Alias: da.
DelegationAssignmentId	Int	No	The Id of the delegation assignment to be removed. Alias: id.

Returns: DirectoryGeneralResponse

Notifications and audit

Configure workflow notifications and read the audit trail.

Get-NotificationAction · Get-Notification · Add-Notification · Get-AuditEvent · Add-AuditEvent

Get-NotificationAction

Gets the list of Actions for which Notifications can be configured.

The Get-NotificationAction cmdlet gets the list of Actions for which Notifications can be configured.

```
Get-NotificationAction
```

This cmdlet takes no parameters.

Returns: Dictionary<String, Dictionary<String, String>>

Get-Notification

Gets the configured notifications that match the specified criteria.

The Get-Notification cmdlet gets the configured notifications that match the specified criteria.

```
Get-Notification [-ProviderName <String>] [-RepositoryKey <String>]
                 [-RepositoryName <String>] [-To <String>] [-RoleGuid <String>]
```

Parameter	Type	Required	Description
ProviderName	String	No	-
RepositoryKey	String	No	-
RepositoryName	String	No	-
To	String	No	-
RoleGuid	String	No	-

Returns: Dictionary<String, NotificationConfiguration>

Add-Notification

Enables notifications for the specified operation for members of the specified role.

The Add-Notification cmdlet Enables notifications for the specified operation for members of the specified role.

```
Add-Notification [-ProviderName <String>] [-RepositoryKey <String>]
  [-RepositoryName <String>] [-From <String>] [-To <String>] [-Subject <String>]
  [-Message <String>] [-Action <String>]
```

Parameter	Type	Required	Description
ProviderName	String	No	-
RepositoryKey	String	No	-
RepositoryName	String	No	-
From	String	No	-
To	String	No	-
Subject	String	No	-
Message	String	No	-
Action	String	No	-

Get-AuditEvent

Gets the events recorded in the UPA audit trail.

The Get-AuditEvent cmdlet gets the events recorded in the UPA audit trail. Each event has TimeStamp, EventType, UserSID, GlobalPolicyID, Comment and GPDomain. Filter the results with Where-Object; the task Read the audit trail shows how to resolve UserSID to an account name.

```
Get-AuditEvent
```

This cmdlet takes no parameters.

Example

```
Get-AuditEvent | Where-Object { $_.GlobalPolicyID -eq $up.GlobalPolicyId } |
  Sort-Object TimeStamp -Descending | Select-Object TimeStamp, EventType, Comment
```

Add-AuditEvent

Adds an event to the UPA audit trail.

The Add-AuditEvent cmdlet adds an event to the UPA audit trail.

```
Add-AuditEvent -Event <AuditEvent>
```

Parameter	Type	Required	Description
Event	AuditEvent	Yes	Accepts pipeline input.

Group Policy Objects in Active Directory

Work directly with GPOs, their links and their permissions.

[Get-HapiGPO](#) · [New-GPONameFilter](#) · [Add-HapiGPO](#) · [Remove-HapiGPO](#) · [Set-HapiGPOProperties](#) · [Get-HapiGPOSettings](#) · [Set-HapiGPOSettings](#) · [Get-HapiGPOReport](#) · [Get-HapiGPORSOPReport](#) · [Get-HapiGPOLinks](#) · [Add-HapiGPOLink](#) · [Update-HapiGPOLink](#) · [Remove-HapiGPOLink](#) · [Get-HapiGPOPermissions](#) · [Add-HapiGPOPermission](#) · [Update-HapiGPOPermissions](#) · [Remove-HapiGPOPermission](#) · [Set-GPOSecurityFiltering](#)

Get-HapiGPO

Gets one or more Group Policy objects from Active Directory.

The Get-HapiGPO cmdlet gets a GPO, or performs a search to retrieve multiple GPOs.

```
Get-HapiGPO [-GPOId <String>] [-MaxResults <Int>] [-Filters <IGpoFilterCriteria[]>]
[-Domain <String>]
```

Parameter	Type	Required	Description
GPOId	String	No	Alias: id.
MaxResults	Int	No	Maximum number of items to return.
Filters	IGpoFilterCriteria[]	No	Returns only results matching the filters.
Domain	String	No	The DNS name of the Active Directory domain.

Returns: GPOEnumResponse, MsftGpo

New-GPONameFilter

Creates a filter that allows you to search for a Group Policy object by name.

The New-GPONameFilter cmdlet creates a filter that allows you to search for a Group Policy object by name. The filter can be used with the Get-HapiGPO cmdlet to search for a Group Policy object. The Domain parameter specifies the DNS name of the Active Directory domain.

```
New-GPONameFilter -Name <String> -FilterType <FilterOperation>
```

Parameter	Type	Required	Description
Name	String	Yes	-
FilterType	FilterOperation	Yes	Values: see Appendix B, FilterOperation.

Returns: IGpoFilterCriteria

Add-HapiGPO

Creates a Group Policy Object in Active Directory.

The Add-HapiGPO cmdlet creates a new Group Policy Object in Active Directory.

```
Add-HapiGPO -Name <String> [-Domain <String>]
```

Parameter	Type	Required	Description
Name	String	Yes	Name for the new GPO.
Domain	String	No	The DNS name of the Active Directory domain.

Returns: MsftGpo

Remove-HapiGPO

Removes a GPO from Active Directory.

The Remove-HapiGPO cmdlet deletes a Group Policy object from Active Directory.

```
Remove-HapiGPO -GPOId <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Set-HapiGPOProperties

Sets the properties of a GPO in Active Directory.

The Set-HapiGPOProperties cmdlet updates the properties of the specified GPO. Properties that can be updated include Display Name, WMI Filter, and Enabled/Disabled Status.

```
Set-HapiGPOProperties -GPOId <String> -Domain <String> [-DisplayName <String>]  
[-WQLFilter <WQLFilter>] [-Status <GpoStatus>] [-Rename] [-SetStatus] [-SetWmi]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.
DisplayName	String	No	-
WQLFilter	WQLFilter	No	-
Status	GpoStatus	No	Values: AllSettingsEnabled, UserSettingsDisabled, ComputerSettingsDisabled, AllSettingsDisabled
Rename	Switch	No	-
SetStatus	Switch	No	-
SetWmi	Switch	No	-

Returns: DirectoryGeneralResponse

Get-HapiGPOSettings

Gets the Policy Settings that are defined in the specified GPO.

The Get-HapiGPOSettings cmdlet gets the Policy Settings that are defined in the specified GPO.

```
Get-HapiGPOSettings -GPOId <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GPOSettings

Set-HapiGPOSettings

Updates the Policy Settings defined in the specified GPO.

The Set-HapiGPOSettings cmdlet updates the Policy Settings defined in the specified GPO.

```
Set-HapiGPOSettings -GPOId <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GPOSettings

Get-HapiGPOReport

Gets the Settings report for the specified GPO.

The Get-HapiGPOReport cmdlet gets the Settings report for the specified GPO.

```
Get-HapiGPOReport -GPOId <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: String

Get-HapiGPORSOPReport

Generates an RSOP report for the specified Computer and/or User.

The Get-HapiGPORSOPReport cmdlet generates an RSOP report for the specified Computer and/or User.

```
Get-HapiGPORSOPReport -Computer <String> -User <String> -Domain <String>
```

Parameter	Type	Required	Description
Computer	String	Yes	-
User	String	Yes	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: String

Get-HapiGPOLinks

Gets the Scopes a GPO is linked to, or the GPOs linked to a specified scope.

The Get-HapiGPOLinks cmdlet gets the Scopes (i.e. Sites, OUs) that a specified GPO is linked to, or gets the GPOs that are linked to the specified Scope.

```
Get-HapiGPOLinks -Domain <String> [-GPOId <String>] [-Location <String>]
```

Parameter	Type	Required	Description
GPOId	String	No	Alias: id.
Location	String	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GpoScopeInfo[]

Add-HapiGPOLink

Links a GPO to a Scope.

The Add-HapiGPOLink cmdlet links a Group Policy object to a Scope, such as an OU or Site, in Active Directory.

```
Add-HapiGPOLink -GPOId <String> -DisplayName <String> -Location <String>  
-Domain <String> [-Enabled] [-Enforced]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
DisplayName	String	Yes	-
Location	String	Yes	-
Enabled	Switch	No	-
Enforced	Switch	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GPOLink

Update-HapiGPOLink

Updates the settings of a GPO Link.

The Update-HapiGPOLink cmdlet sets the Enabled and/or Enforced attributes on a link between a GPO and a Scope (Site or OU).

```
Update-HapiGPOLink -GPOId <String> -Domain <String> -Location <String> [-Enable]
[-Disable] [-Enforce] [-Unenforce]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.
Location	String	Yes	-
Enable	Switch	No	-
Disable	Switch	No	-
Enforce	Switch	No	-
Unenforce	Switch	No	-

Returns: GPOLink

Remove-HapiGPOLink

Removes a GPO Link from Active Directory.

The Remove-HapiGPOLink cmdlet removes a link to a GPO from a Scope (i.e. Site, or OU).

```
Remove-HapiGPOLink -GPOId <String> -Location <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Location	String	Yes	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Get-HapiGPOPermissions

Gets the permissions for the specified GPO.

The Get-HapiGPOPermissions cmdlet gets the permissions for the specified GPO.

```
Get-HapiGPOPermissions -GPOId <String> -Domain <String>
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: MsftGPOPermission[]

Add-HapiGPOPermission

Grants the specified User/Group Permissions on a GPO.

The Add-HapiGPOPermission cmdlet grants the specified User/Group permissions on a GPO, in Active Directory.

```
Add-HapiGPOPermission -GPOId <String> -Domain <String>
[-PermissionType <MsftGPPermissionType>] [-Name <String>]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
PermissionType	MsftGPPermissionType	No	Values: see Appendix B, MsftGPPermissionType. Alias: perm.
Name	String	No	Alias: user,group.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: MsftGPOPermission[]

Update-HapiGPOPermissions

Updates the Permissions for the specified GPO.

The Update-HapiGPOPermissions cmdlet updates the Permissions for the specified GPO by adding/removing permissions for the specified Users/Groups.

```
Update-HapiGPOPermissions -GPOId <String> -Domain <String>
[-ToAdd <MsftGPOPermission[]>] [-ToRemove <MsftGPOPermission[]>]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.
ToAdd	MsftGPOPermission[]	No	-
ToRemove	MsftGPOPermission[]	No	-

Returns: MsftGPOPermission[]

Remove-HapiGPOPermission

Removes the specified permission entry from a GPO's permissions.

The Remove-HapiGPOPermission cmdlet revokes the specified permission on a Group Policy object from Active Directory.

```
Remove-HapiGPOPermission -PermissionType <MsftGPPermissionType> -Domain <String>
[-GPOId <String>] [-Name <String>]
```

Parameter	Type	Required	Description
GPOId	String	No	Alias: id.
PermissionType	MsftGPPermissionType	Yes	Values: see Appendix B, MsftGPPermissionType. Alias: perm.
Name	String	No	Alias: user,group.
Domain	String	Yes	The DNS name of the Active Directory domain.

Set-GPOSecurityFiltering

Sets the security filtering of a GPO in Active Directory.

The Set-GPOSecurityFiltering cmdlet updates the security filtering of the specified GPO.

```
Set-GPOSecurityFiltering -GPOId <String> -Domain <String>
[-Operation <SecurityFilteringOperation>] [-Sids <String[]>]
[-Accounts <SecurityPrincipalInfo[]>]
```

Parameter	Type	Required	Description
GPOId	String	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.
Operation	SecurityFilteringOperation	No	The Operation to perform (Add/Remove/Replace) Values: Replace, Add, Remove
Sids	String[]	No	The SIDs of the users/groups to modify security filtering for.
Accounts	SecurityPrincipalInfo[]	No	An array of SecurityPrincipalInfo objects containing the SIDs of the users/groups to modify security filtering for.

Returns: DirectoryGeneralResponse

Active Directory

Query domains, users, groups and computers, and manage users and groups.

[Get-ADDomains](#) · [Set-DomainCredential](#) · [Get-ADDomainControllers](#) · [Get-ADContainers](#) · [Get-ADObjects](#) · [Get-ADUsers](#) · [Get-ADGroups](#) · [Get-ADComputers](#) · [Get-PosixUsers](#) · [New-LdapFilter](#) · [New-LdapPropertyValue](#) · [New-User](#) · [Set-User](#) · [Remove-User](#) · [New-Group](#) · [Set-Group](#) · [Remove-Group](#) · [Add-GroupMember](#) · [Remove-GroupMember](#)

Get-ADDomains

Gets the set of Active Directory domains that can be managed by UPA.

The Get-ADDomains cmdlet gets the set of domains that can be managed by UPA, along with the usernames of any alternate credentials that are used to manage those domains. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-ADDomains
```

This cmdlet takes no parameters.

Returns: ADDomainEnumResponse

Set-DomainCredential

Specifies credentials to manage a domain.

The Set-DomainCredential cmdlet specifies the credentials UPA uses to perform operations in an Active Directory domain. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Set-DomainCredential -Domain <String> [-ServerOrIpAddress <String>]
    [-UseAlternateCredentialForRead <Boolean>]
    [-UseAlternateCredentialForWrite <Boolean>] [-Username <String>]
    [-Password <String>]
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
ServerOrIpAddress	String	No	(Optional) The IP Address or DNS name of a domain controller to use when performing operations for this domain.
UseAlternateCredentialForRead	Boolean	No	Indicates whether to use the specified credentials for read operations for this domain. (Default = \$True).
UseAlternateCredentialForWrite	Boolean	No	Indicates whether to use the specified credentials for write operations for this domain. (Default = \$True).
Username	String	No	The username (i.e. DOMAIN\USERNAME) to use for read and/or write operations for this domain, as specified by UseAlternateCredentialsForRead and UseAlternateCredentialsForWrite. If no username is specified, and UseAlternateCredentialsForRead or UseAlternateCredentialsForWrite is \$True, the gateway will use its IIS service account to access the specified domain.
Password	String	No	The password for the account specified in the Username parameter. If a Username is specified, a password must also be specified.

Returns: DirectoryGeneralResponse

Get-ADDomainControllers

Gets the list of Active Directory Domain Controllers for the specified domain.

The Get-ADDomainControllers cmdlet gets the list of Domain Controllers for the specified domain.

```
Get-ADDomainControllers -Domain <String> [-BaseContainer <String>]
[-InfoLevel <InformationLevel>] [-SkipCount <Int>] [-MaxResults <Int>]
[-PageSize <Int>] [-ResumePoint <String>] [-Filters <IFilterCriteria[]>]
```

Parameter	Type	Required	Description
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
InfoLevel	InformationLevel	No	Which properties to return. Values: Default, Basic, Details, Membership, Photo, Status, Role, PrimaryGroup, All
SkipCount	Int	No	Indicates the number of items to skip.
MaxResults	Int	No	Maximum number of items to return.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.
Filters	IFilterCriteria[]	No	Returns only results matching the filters.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DCEnumResponse

Get-ADContainers

Gets one or more Active Directory containers (Domain, Sites, OUs, and Containers).

The Get-ADContainers cmdlet performs a search to retrieve multiple containers.

```
Get-ADContainers -Domain <String> [-IncludeContainers] [-IncludeLinkedGPOs]
[-BaseContainer <String>] [-LdapProperties <String[]>] [-MaxResults <Int>]
[-PageSize <Int>] [-ResumePoint <String>]
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
IncludeContainers	Switch	No	Indicates whether just Domains, Sites, and OUs should be returned, or if Containers should also be included in the search.
IncludeLinkedGPOs	Switch	No	Indicates whether to include which GPOs are linked to the containers.
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=... If null, then the whole domain will be searched.
LdapProperties	String[]	No	Which additional LDAP properties to return.
MaxResults	Int	No	Maximum number of items to return. Set to 0 to return all items.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.

Parameter	Type	Required	Description
ResumePoint	String	No	Indicates the resume point for a multiple page search.

Returns: GroupEnumResponse

Get-ADObjects

Gets Active Directory objects of one or more types.

The Get-ADObjects cmdlet gets Active Directory objects of the types named in ObjectTypes, such as users, groups, computers or OUs, or performs a search to retrieve multiple objects. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-ADObjects -Domain <String> -ObjectTypes <ADObjectTypes> [-BaseContainer <String>]
[-InfoLevel <InformationLevel>] [-SkipCount <Int>] [-MaxResults <Int>]
[-PageSize <Int>] [-ResumePoint <String>] [-Filters <IFilterCriteria[]>]
```

Parameter	Type	Required	Description
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
InfoLevel	InformationLevel	No	Which properties to return. Values: Default, Basic, Details, Membership, Photo, Status, Role, PrimaryGroup, All
SkipCount	Int	No	Indicates the number of items to skip.
MaxResults	Int	No	Maximum number of items to return.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.
Filters	IFilterCriteria[]	No	Returns only results matching the filters.
Domain	String	Yes	The DNS name of the Active Directory domain.
ObjectTypes	ADObjectTypes	Yes	The types of objects to retrieve. Values: Default, Users, Groups, OUs, Computers, DomainControllers, All, Custom

Returns: GroupEnumResponse

Get-ADUsers

Gets one or more Active Directory users.

The Get-ADUsers cmdlet gets a user, or performs a search to retrieve multiple users. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-ADUsers -Domain <String> [-BaseContainer <String>] [-InfoLevel <InformationLevel>]
[-SkipCount <Int>] [-MaxResults <Int>] [-PageSize <Int>] [-ResumePoint <String>]
[-Filters <IFilterCriteria[]>] [-LdapProperties <String[]>]
```

Parameter	Type	Required	Description
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
InfoLevel	InformationLevel	No	Which properties to return. Values: Default, Basic, Details, Membership, Photo, Status, Role, PrimaryGroup, All
SkipCount	Int	No	Indicates the number of items to skip.
MaxResults	Int	No	Maximum number of items to return.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.
Filters	IFilterCriteria[]	No	Returns only results matching the filters.
LdapProperties	String[]	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: UserEnumResponse

Get-ADGroups

Gets one or more Active Directory groups.

The Get-ADGroups cmdlet gets a group, or performs a search to retrieve multiple groups. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-ADGroups -Domain <String> [-BaseContainer <String>] [-InfoLevel <InformationLevel>]
[-SkipCount <Int>] [-MaxResults <Int>] [-PageSize <Int>] [-ResumePoint <String>]
[-Filters <IFilterCriteria[]>]
```

Parameter	Type	Required	Description
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
InfoLevel	InformationLevel	No	Which properties to return. Values: Default, Basic, Details, Membership, Photo, Status, Role, PrimaryGroup, All
SkipCount	Int	No	Indicates the number of items to skip.
MaxResults	Int	No	Maximum number of items to return.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.
Filters	IFilterCriteria[]	No	Returns only results matching the filters.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GroupEnumResponse

Get-ADComputers

Gets one or more Active Directory computers.

The Get-ADComputers cmdlet gets a computer, or performs a search to retrieve multiple computers. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-ADComputers -Domain <String> [-BaseContainer <String>]
[-InfoLevel <InformationLevel>] [-SkipCount <Int>] [-MaxResults <Int>]
[-PageSize <Int>] [-ResumePoint <String>] [-Filters <IFilterCriteria[]>]
```

Parameter	Type	Required	Description
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
InfoLevel	InformationLevel	No	Which properties to return. Values: Default, Basic, Details, Membership, Photo, Status, Role, PrimaryGroup, All
SkipCount	Int	No	Indicates the number of items to skip.
MaxResults	Int	No	Maximum number of items to return.
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.
Filters	IFilterCriteria[]	No	Returns only results matching the filters.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: ComputerEnumResponse

Get-PosixUsers

Gets posix attributes for one or more Active Directory users.

The Get-PosixUsers cmdlet gets the posix attributes of a user, or performs a search to retrieve posix attributes for multiple users. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-PosixUsers [-IncludePosixOnly <Boolean>] [-BaseContainer <String>] [-PageSize <Int>]
[-ResumePoint <String>] [-FilterProperty <String>] [-FilterValue <String>]
[-FilterOperation <FilterOperation>] [-Group <String>] [-Domain <String>]
[-AgentId <Int>]
```

Parameter	Type	Required	Description
IncludePosixOnly	Boolean	No	-
BaseContainer	String	No	Distinguished name of the container to search in. i.e. CN=Users,DC=...
PageSize	Int	No	Number of results to return for each page. Use ResumePoint to retrieve subsequent pages.
ResumePoint	String	No	Indicates the resume point for a multiple page search.

Parameter	Type	Required	Description
FilterProperty	String	No	-
FilterValue	String	No	-
FilterOperation	FilterOperation	No	Values: see Appendix B, FilterOperation.
Group	String	No	-
Domain	String	No	-
AgentId	Int	No	-

Returns: UserEnumResponse

New-LdapFilter

Creates a filter that allows you to search for a User, Group, or Computer based on its properties.

The New-LdapFilter cmdlet creates a filter that allows you to search for a User, Group, or Computer based on its properties. The filter can be used with the Get-ADUsers, Get-ADGroups, Get-ADComputers, and Get-PosixUsers cmdlets to search for objects that match the filter. The Domain parameter specifies the DNS name of the Active Directory domain.

```
New-LdapFilter -Property <String> -Value <String> -FilterType <FilterOperation>
```

Parameter	Type	Required	Description
Property	String	Yes	-
Value	String	Yes	-
FilterType	FilterOperation	Yes	Values: see Appendix B, FilterOperation.

Returns: IFilterCriteria

New-LdapPropertyValue

Creates an LdapPropertyValue object.

The New-LdapPropertyValue cmdlet creates an LdapPropertyValue object that can be used to update a user or group property using the Set-User or Set-Group cmdlets. LdapPropertyValue objects can also be created using the New-Object powershell cmdlet.

```
New-LdapPropertyValue -PropertyName <String> -JsonValue <String>
[-PropertyType <String>] [-Value <Object>]
```

Parameter	Type	Required	Description
PropertyName	String	Yes	The LDAP property name. Accepts pipeline input.
PropertyType	String	No	(Optional) The type of the property value.
JsonValue	String	Yes	-
Value	Object	No	-

New-User

Creates a new Active Directory user.

The New-User cmdlet creates a new user in Active Directory. The Domain parameter specifies the DNS name of the Active Directory domain.

```
New-User -SamAccountName <String> -Password <String> -Domain <String>
  [-DisplayName <String>] [-Email <String>] [-GivenName <String>] [-Surname <String>]
  [-ContainerDN <String>]
```

Parameter	Type	Required	Description
SamAccountName	String	Yes	The SAM Account Name (pre-Windows 2000 Login Name) for the user.
DisplayName	String	No	The display name for the user. If a display name is not specified, the SamAccountName will be used as the display name.
Email	String	No	The email address for the user.
GivenName	String	No	The given name of the user.
Surname	String	No	The surname of the user.
Password	String	Yes	The password for the new user.
ContainerDN	String	No	The Distinguished name (i.e. OU=Test,DC=...) of the OU to create the new user account in.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryAddUsersResponse

Set-User

Updates an Active Directory user.

The Set-User cmdlet updates the properties of an Active Directory user. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Set-User -User <ADUserEntry> -Properties <LdapPropertyValue[]> -Domain <String>

Set-User -UserSID <String> -Properties <LdapPropertyValue[]> -Domain <String>
```

Parameter	Type	Required	Description
User	ADUserEntry	Yes	An ADUserEntry containing the SID for the User to update.
Properties	LdapPropertyValue[]	Yes	An array of LdapPropertyValue objects specifying the properties to update.
Domain	String	Yes	The DNS name of the Active Directory domain.
UserSID	String	Yes	The SID (i.e. S-1-...) of the User to update.

Returns: DirectoryUpdateUserResponse

Remove-User

Deletes a user from Active Directory.

The Remove- Cmdlet deletes a user account from Active Directory. The Domain parameter specifies the DNS name of the Active Directory domain. The Group and Member can be specified either as SIDs in SDDL format, or as Distinguished Names.

```
Remove-User -User <ADUserEntry> -Domain <String>
```

```
Remove-User -UserSID <String> -Domain <String>
```

Parameter	Type	Required	Description
User	ADUserEntry	Yes	An ADUserEntry object returned from Get-ADUsers, representing the user to delete. Accepts pipeline input.
Domain	String	Yes	-
UserSID	String	Yes	The SID of the Active Directory user to delete, in SDDL format.

Returns: DirectoryGeneralResponse

New-Group

Creates a new Active Directory group.

The New-Group cmdlet creates a new group in Active Directory. The Domain parameter specifies the DNS name of the Active Directory domain.

```
New-Group -SamAccountName <String> -Domain <String> [-DisplayName <String>]  
[-Description <String>] [-GroupType <WindowsGroupType>] [-ContainerDN <String>]
```

Parameter	Type	Required	Description
SamAccountName	String	Yes	The SAMAccountName for the new group.
DisplayName	String	No	(Optional) The display name for the new group. If a display name is not specified, the SAMAccountName will be used as the display name.
Description	String	No	(Optional) The description property for the new group.
GroupType	WindowsGroupType	No	The type of the group. If no group type is specified, a global security group will be created. Values: see Appendix B, WindowsGroupType.
ContainerDN	String	No	The Distinguished Name (i.e. OU=Test,DC=...) of the Active Directory OU to create the group in.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryAddGroupsResponse

Set-Group

Updates an Active Directory group.

The Set-Group cmdlet updates the properties of an Active Directory group. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Set-Group -Group <ADGroupEntry> -Properties <LdapPropertyValue[]> -Domain <String>

Set-Group -GroupSID <String> -Properties <LdapPropertyValue[]> -Domain <String>
```

Parameter	Type	Required	Description
Group	ADGroupEntry	Yes	An ADGroupEntry (possibly returned from Get-ADGroups) containing the SID of the Group to update.
Properties	LdapPropertyValue[]	Yes	An array of LdapPropertyValue objects specifying the properties to update.
Domain	String	Yes	The DNS name of the Active Directory domain.
GroupSID	String	Yes	The SID (i.e. S-1-...) of the Group to update.

Returns: DirectoryUpdateGroupResponse

Remove-Group

Deletes a group from Active Directory.

The Remove- Cmdlet deletes a group from Active Directory. The Domain parameter specifies the DNS name of the Active Directory domain. The Group can be specified as a SID in SDDL format, or by passing in an ADGroupEntry object returned from Get-ADGroups.

```
Remove-Group -Group <ADGroupEntry> -Domain <String>

Remove-Group -GroupSID <String> -Domain <String>
```

Parameter	Type	Required	Description
Group	ADGroupEntry	Yes	ADGroupEntry representing the group. The ADGroupEntry must include the GroupSID property. Accepts pipeline input.
Domain	String	Yes	The DNS name of the Active Directory domain.
GroupSID	String	Yes	The SID of the group to delete in SDDL format.

Returns: DirectoryGeneralResponse

Add-GroupMember

Adds a member to an Active Directory group.

The Add-GroupMember Cmdlet adds a member to an Active Directory group. The Domain parameter specifies the DNS name of the Active Directory domain. The Group and Member can be specified either as SIDs in SDDL format, or as Distinguished Names.

```
Add-GroupMember -Group <String> -Member <String> -Domain <String>
```

Parameter	Type	Required	Description
Group	String	Yes	The group to add a member to. You can specify the Group either as a SID in SDDL format, or as a distinguished name.
Member	String	Yes	Identifies the user or group to add as a group member. You can specify the Member either as a SID in SDDL format, or as a distinguished name.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryGeneralResponse

Remove-GroupMember

Removes a member from a group in Active Directory.

The Remove-GroupMember cmdlet removes a group member from an Active Directory group. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Remove-GroupMember -Group <String> -Member <String> -Domain <String>
```

Parameter	Type	Required	Description
Group	String	Yes	The SID (i.e. S-1-...) or Distinguished Name (i.e. CN=...) of the group to remove a member from.
Member	String	Yes	The SID (i.e. S-1-...) or Distinguished Name (i.e. CN=...) of the member to remove.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryGeneralResponse

WMI filters

Manage WMI filters in Active Directory and in the UPA repository.

Get-WMIFilter · New-WMIFilter · Set-WMIFilter · Remove-WMIFilter · Test-WMIFilter · Get-WMINamespaces · Get-RepWMIFilter · Set-RepWMIFilter · Remove-RepWMIFilter

Get-WMIFilter

Gets one or more WMI Filters from Active Directory.

The Get-WMIFilter cmdlet gets a WMI Filter, or performs a search to retrieve multiple WMI Filters.

```
Get-WMIFilter -Domain <String> [-MaxResults <Int>] [-Filters <IGpoFilterCriteria[]>]
[-FilterId <Guid>]
```

Parameter	Type	Required	Description
MaxResults	Int	No	Maximum number of items to return.

Parameter	Type	Required	Description
Filters	IGpoFilterCriteria[]	No	Returns only results matching the filters.
FilterId	Guid	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: WQLFilter[]

New-WMIFilter

Creates a new WMI Filter in Active Directory.

The New-WMIFilter cmdlet creates a WMI Filter in Active Directory. This WMI filter can then be assigned to a GPO.

```
New-WMIFilter -Name <String> -Domain <String> [-Description <String>]
[-Namespace <String>] [-Query <String>]
```

Parameter	Type	Required	Description
Name	String	Yes	-
Description	String	No	-
Namespace	String	No	-
Query	String	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: WQLFilter

Set-WMIFilter

Updates a WMI Filter in Active Directory.

The Set-WMIFilter cmdlet updates a WMI Filter in Active Directory. The updated WMI Filter settings will apply to all GPOs that use the specified WMI Filter.

```
Set-WMIFilter -Filter <WQLFilter> -Domain <String>
```

Parameter	Type	Required	Description
Filter	WQLFilter	Yes	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: WQLFilter

Remove-WMIFilter

Removes a WMI Filter from Active Directory.

The Remove-WMIFilter cmdlet deletes a WMI Filter from Active Directory.

```
Remove-WMIFilter -FilterId <Guid> -Domain <String>
```

Parameter	Type	Required	Description
FilterId	Guid	Yes	Alias: id.
Domain	String	Yes	The DNS name of the Active Directory domain.

Test-WMIFilter

Runs a WMI Filter to determine whether the filter is valid.

The Test-WMIFilter cmdlet runs a WMI Filter against the gateway server, in order to determine whether the filter is valid.

```
Test-WMIFilter -Filter <WQLFilter> [-Domain <String>]
```

Parameter	Type	Required	Description
Filter	WQLFilter	Yes	-
Domain	String	No	-

Returns: DirectoryGeneralResponse

Get-WMINamespaces

Gets the set of WMI Namespaces.

The Get-WMINamespaces cmdlet gets the WMI Namespaces that can be used in WMI Filters for GPOs.

```
Get-WMINamespaces
```

This cmdlet takes no parameters.

Returns: String[]

Get-RepWMIFilter

Retrieves one or more WMI filters from storage.

The Get-RepWMIFilter cmdlet retrieves one or more WMI filters from the repository.

```
Get-RepWMIFilter -Domain <String> [-Id <Guid>]
```

Parameter	Type	Required	Description
Domain	String	Yes	Returns only results matching the filters.
Id	Guid	No	-

Returns: List<WQLFilter>

Set-RepWMIFilter

Creates or Updates a WMI Filter in the Repository.

The Set-RepWMIFilter cmdlet Creates or Updates a WMI Filter in the Repository.

```
Set-RepWMIFilter [-Filter <WQLFilter>] [-Domain <String>] [-Name <String>]
[-Description <String>] [-Namespace <String>] [-Query <String>]
```

Parameter	Type	Required	Description
Filter	WQLFilter	No	Accepts pipeline input.
Domain	String	No	-
Name	String	No	-
Description	String	No	-
Namespace	String	No	-
Query	String	No	-

Returns: WQLFilter

Remove-RepWMIFilter

Removes a WMI Filter from the Repository.

The Remove-RepWMIFilter cmdlet removes the specified WMI filter from the Repository.

```
Remove-RepWMIFilter [-Filter <WQLFilter>] [-Id <Guid>] [-Domain <String>]
```

Parameter	Type	Required	Description
Filter	WQLFilter	No	Accepts pipeline input.
Id	Guid	No	-
Domain	String	No	-

Administrative templates

Manage the ADMX and ADML files available to Universal Policies.

[Get-AdmFiles](#) · [Get-UnAddedADMFiles](#) · [New-AdmFile](#) · [Remove-AdmFile](#)

Get-AdmFiles

Gets ADMX definitions for policy settings.

The Get-AdmFiles cmdlet retrieves the ADMX policy setting definitions for settings that can be added to Universal Policies.

```
Get-AdmFiles [-LanguageId <String>]
```

Parameter	Type	Required	Description
LanguageId	String	No	Alias: lang.

Returns: String

Get-UnAddedADMFiles

Get ADMX or ADML definition files from the Central Store that have not been added to the DB for use in UPs.

The Get-UnAddedADMFiles cmdlet gets ADMX or ADML definition files from the Central Store that have not been added for use in Universal Policies.

```
Get-UnAddedADMFiles [-OnlyADMX] [-OnlyADML] [-LanguageId <String>]
```

Parameter	Type	Required	Description
OnlyADMX	Switch	No	Alias: admx.
OnlyADML	Switch	No	Alias: adml.
LanguageId	String	No	Alias: lang.

Returns: String

New-AdmFile

Adds an ADMX or ADML definitions for policy settings.

The New-AdmFile cmdlet adds an ADMX or ADML policy setting definition for settings that can be added to Universal Policies.

```
New-AdmFile [-FilePath <String>] [-FileName <String>] [-LanguageId <String>]
```

Parameter	Type	Required	Description
FilePath	String	No	Accepts pipeline input. Alias: path.
FileName	String	No	Alias: name.
LanguageId	String	No	-

Returns: String

Remove-AdmFile

Removes an ADMX or ADML file from the templates available to Universal Policies.

The Remove-AdmFile cmdlet removes an administrative template (ADMX) file, or a language (ADML) file when LanguageId is specified, from the templates available to Universal Policies.

```
Remove-AdmFile -FileName <String> [-LanguageId <String>]
```

Parameter	Type	Required	Description
FileName	String	Yes	-
LanguageId	String	No	-

Migration maps

Manage the maps that translate paths and security principals when a policy moves between domains.

[Get-MigrationMap](#) · [Set-MigrationMap](#) · [Remove-MigrationMap](#)

Get-MigrationMap

Gets one or more Migration Maps from storage.

The Get-MigrationMap cmdlet retrieves one or more Migration Maps from storage. A Migration Map is used to migrate Universal Policies between different domains.

```
Get-MigrationMap -SourceDomain <String> -TargetDomain <String>
```

Parameter	Type	Required	Description
SourceDomain	String	Yes	The name of the source domain for the migration map.
TargetDomain	String	Yes	The name of the target domain for the migration map.

Returns: List<MigrationMap>

Set-MigrationMap

Creates or updates a Migration Map in storage.

The Set-MigrationMap cmdlet creates or updates a Migration Map in storage. A Migration Map is used to migrate Universal Policies between domains.

```
Set-MigrationMap [-Map <MigrationMap>]
```

Parameter	Type	Required	Description
Map	MigrationMap	No	Accepts pipeline input.

Returns: MigrationMap

Remove-MigrationMap

Removes one or more Migration Maps from storage.

The Remove-MigrationMap cmdlet removes one or more Migration Maps from storage. Migration Maps are used to migrate Universal Policies between domains.

```
Remove-MigrationMap [-Map <MigrationMap>] [-SourceDomain <String>]
[-TargetDomain <String>]
```

Parameter	Type	Required	Description
Map	MigrationMap	No	Accepts pipeline input.
SourceDomain	String	No	-
TargetDomain	String	No	-

Returns: ItemsResponse<PolicyLinkInfo>

IP Security

Manage IP Security policies, filter lists and filter actions in Active Directory.

[Get-IPSecPolicy](#) · [Add-IPSecPolicy](#) · [Get-IPSecFilterList](#) · [Add-IPSecFilterList](#) · [Set-IPSecFilterList](#) · [Remove-IPSecFilterList](#) · [Get-IPSecFilterAction](#) · [Add-IPSecFilterAction](#) · [Set-IPSecFilterAction](#) · [Remove-IPSecFilterAction](#)

Get-IPSecPolicy

Gets one or more IP Security policies from Active Directory.

The Get-IPSecPolicy cmdlet gets an IP Security policy, or performs a search to retrieve multiple policies. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-IPSecPolicy -Domain <String> [-MaxResults <Int>] [-Filters <IGpoFilterCriteria[]>]
[-PolicyId <Guid>]
```

Parameter	Type	Required	Description
MaxResults	Int	No	Maximum number of items to return.
Filters	IGpoFilterCriteria[]	No	Returns only results matching the filters.
PolicyId	Guid	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Add-IPSecPolicy

Creates an IP Security policy in Active Directory.

The Add-IPSecPolicy cmdlet creates an IP Security policy in the specified domain.

```
Add-IPSecPolicy -Policy <IPSecPolicy> -Domain <String>
```

Parameter	Type	Required	Description
Policy	IPSecPolicy	Yes	The IPSecPolicy to create.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: IPSecPolicy

Get-IPSecFilterList

Gets one or more IP Security filter lists from Active Directory.

The Get-IPSecFilterList cmdlet gets an IP Security filter list, or performs a search to retrieve multiple filter lists. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-IPSecFilterList -Domain <String> [-MaxResults <Int>]
[-Filters <IGpoFilterCriteria[]>] [-PolicyId <Guid>]
```

Parameter	Type	Required	Description
MaxResults	Int	No	Maximum number of items to return.
Filters	IGpoFilterCriteria[]	No	Returns only results matching the filters.
PolicyId	Guid	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Add-IPSecFilterList

Creates an IP Security filter list in Active Directory.

The Add-IPSecFilterList cmdlet creates an IP Security filter list in the specified domain.

```
Add-IPSecFilterList -FilterList <IPFilterList> -Domain <String>
```

Parameter	Type	Required	Description
FilterList	IPFilterList	Yes	The filter list to create.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: IPFilterList

Set-IPSecFilterList

Updates an IP Security filter list in Active Directory.

The Set-IPSecFilterList cmdlet updates an existing IP Security filter list in the specified domain.

```
Set-IPSecFilterList -FilterList <IPFilterList> -Domain <String>
```

Parameter	Type	Required	Description
FilterList	IPFilterList	Yes	The filter List to create.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: IPFilterList

Remove-IPSecFilterList

Removes an IP Security filter list from Active Directory.

The Remove-IPSecFilterList cmdlet removes an IP Security filter list, identified by the object or by ListId, from the specified domain.

```
Remove-IPSecFilterList -Domain <String> [-FilterList <IPFilterList>] [-ListId <String>]
```

Parameter	Type	Required	Description
FilterList	IPFilterList	No	The filter List to delete.
ListId	String	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryGeneralResponse

Get-IPSecFilterAction

Gets one or more IP Security filter actions from Active Directory.

The Get-IPSecFilterAction cmdlet gets an IP Security filter action, or performs a search to retrieve multiple filter actions. The Domain parameter specifies the DNS name of the Active Directory domain.

```
Get-IPSecFilterAction -Domain <String> [-MaxResults <Int>]
[-Filters <IGpoFilterCriteria[]>] [-PolicyId <Guid>]
```

Parameter	Type	Required	Description
MaxResults	Int	No	Maximum number of items to return.
Filters	IGpoFilterCriteria[]	No	Returns only results matching the filters.
PolicyId	Guid	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Add-IPSecFilterAction

Creates an IP Security filter action in Active Directory.

The Add-IPSecFilterAction cmdlet creates an IP Security filter action in the specified domain.

```
Add-IPSecFilterAction -FilterAction <IPFilterAction> -Domain <String>
```

Parameter	Type	Required	Description
FilterAction	IPFilterAction	Yes	The filter action to create.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: IPFilterAction

Set-IPSecFilterAction

Updates an IP Security filter action in Active Directory.

The Set-IPSecFilterAction cmdlet updates an existing IP Security filter action in the specified domain.

```
Set-IPSecFilterAction -FilterAction <IPFilterAction> -Domain <String>
```

Parameter	Type	Required	Description
FilterAction	IPFilterAction	Yes	The filter action to create.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: IPFilterAction

Remove-IPSecFilterAction

Removes an IP Security filter action from Active Directory.

The Remove-IPSecFilterAction cmdlet removes an IP Security filter action, identified by the object or by ActionId, from the specified domain.

```
Remove-IPSecFilterAction -Domain <String> [-FilterAction <IPFilterAction>]
[-ActionId <String>]
```

Parameter	Type	Required	Description
FilterAction	IPFilterAction	No	The filter action to delete.
ActionId	String	No	-
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryGeneralResponse

Software installation

Manage software installation packages and categories.

Get-SWInstallPackage · Get-SWInstallSettingFromPackage · Get-SoftwareInstallCategory · New-SoftwareInstallCategory · Set-SoftwareInstallCategory · Remove-SoftwareInstallCategory

Get-SWInstallPackage

Gets one or more software install packages from Active Directory.

The Get-SWInstallPackage cmdlet gets the software install packages defined in the specified GPO.

```
Get-SWInstallPackage -Domain <String> [-UserPolicies] [-GpoId <Guid>]
[-ThatUpgrade <Guid>]
```

Parameter	Type	Required	Description
UserPolicies	Switch	No	Returns packages defined in user policies.
Gpoid	Guid	No	The ID of the GPO to retrieve packages from.
Domain	String	Yes	The DNS name of the Active Directory domain.
ThatUpgrade	Guid	No	Gets packages that upgrade the specified package.

Get-SWInstallSettingFromPackage

Reads an MSI package and returns the settings for a software installation item.

The Get-SWInstallSettingFromPackage cmdlet reads an MSI package, with any transforms you specify, and returns the settings for a software installation item in the specified domain.

```
Get-SWInstallSettingFromPackage -Domain <String> -Path <String> [-Transforms <String[]>]
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
Path	String	Yes	The path to the MSI package.
Transforms	String[]	No	Specify the path to one or more .mst transforms to apply to the MSI package."

Returns: GetMsiSummaryResponse[]

Get-SoftwareInstallCategory

Gets one or more Software Install Categories from Active Directory.

The Get-SoftwareInstallCategory cmdlet gets the Software Install Categories for the specified domain.

```
Get-SoftwareInstallCategory -Domain <String>
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.

New-SoftwareInstallCategory

Adds a new Software Install Categories.

The New-SoftwareInstallCategory cmdlet adds a Software Install Category for the specified domain.

```
New-SoftwareInstallCategory -Domain <String> -Name <String>
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
Name	String	Yes	The name for the new software install category.

Returns: AddSoftwareInstallCategoryResponse

Set-SoftwareInstallCategory

Updates a Software Install Category.

The Set-SoftwareInstallCategory cmdlet updates a Software Install Category for the specified domain.

```
Set-SoftwareInstallCategory -Domain <String> -Category <SoftwareInstallCategory>
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
Category	SoftwareInstallCategory	Yes	The software install category to update.

Returns: AddSoftwareInstallCategoryResponse

Remove-SoftwareInstallCategory

Removes a Software Install Category from Active Directory.

The Remove-SoftwareInstallCategory cmdlet removes the Software Install Category with the specified Id from the specified domain.

```
Remove-SoftwareInstallCategory -Domain <String> -Id <Guid>
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
Id	Guid	Yes	The id of the software install category to remove.

Returns: DirectoryGeneralResponse

Lookups for policy settings

Retrieve the values that certain policy settings and preference items are built from.

[Get-CertTemplate](#) · [Get-TrustedRoot](#) · [Get-CentralAccessPolicy](#) · [Get-PredefinedFirewallRules](#) · [Get-ODBC](#) · [Get-Services](#) · [Get-AppPackages](#) · [Get-DeviceInfo](#)

Get-CertTemplate

Gets one or more certificate templates from the certificate authority.

The Get-CertTemplate cmdlet gets a list of the Certificate Templates.

```
Get-CertTemplate -Domain <String> [-Usage <String>] [-IncludeFlags <Long>]
[-ExcludeFlags <Long>]
```

Parameter	Type	Required	Description
Usage	String	No	Only return templates that contain the specified certificate use OID
IncludeFlags	Long	No	Only return templates that contain the specified flags
ExcludeFlags	Long	No	Only return templates that do NOT contain the specified flags
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: GetCertificateTemplateResponse

Get-TrustedRoot

Gets one or more Trusted Root Certificate authorities.

The Get-TrustedRoot cmdlet gets a list of the Trusted Root Certificate authorities.

```
Get-TrustedRoot [-AllDetails] [-Intermediate] [-IncludeExpired] [-Domain <String>]
```

Parameter	Type	Required	Description
AllDetails	Switch	No	Return the full X509Certificate for each CA.
Intermediate	Switch	No	Return Intermediate Certification authorities.
IncludeExpired	Switch	No	Includes certificates that are expired or otherwise invalid.
Domain	String	No	The DNS name of the Active Directory domain.

Returns: GetTrustedRootCAResponse

Get-CentralAccessPolicy

Gets one or more Central Access Policy from Active Directory.

The Get-CentralAccessPolicy cmdlet gets the Central Access Policies for the specified domain.

```
Get-CentralAccessPolicy -Domain <String>
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.

Get-PredefinedFirewallRules

Gets the predefined Windows Firewall rules for use in firewall policy settings.

The Get-PredefinedFirewallRules cmdlet gets the predefined Windows Firewall rules, optionally limited to one rule group. Specify Inbound, Outbound, or both. The result lists the rules in Items and their groups in InboundCategories and OutboundCategories.

```
Get-PredefinedFirewallRules -Domain <String> [-Group <String>] [-Inbound] [-Outbound]
```

Parameter	Type	Required	Description
Domain	String	Yes	The DNS name of the Active Directory domain.
Group	String	No	(Optional) The rule group name.
Inbound	Switch	No	Include inbound rules.
Outbound	Switch	No	Include outbound rules.

Returns: An object with Items, InboundCategories and OutboundCategories.

Example

```
Get-PredefinedFirewallRules -Domain 'contoso.com' -Inbound
```

Get-ODBC

Gets the ODBC drivers and data sources for use in Data Sources preference items.

The Get-ODBC cmdlet gets the ODBC drivers and data source names (DSNs) available, for use in Data Sources preference items. The result lists them in Drivers and DSNs.

```
Get-ODBC
```

This cmdlet takes no parameters.

Returns: An object with Drivers and DSNs.

Example

```
(Get-ODBC).DSNs
```

Get-Services

Gets the Windows services on a server for use in Services settings and preference items.

The Get-Services cmdlet gets the Windows services on the specified server, for use in System Services settings and Services preference items. The result lists them in Items.

```
Get-Services [-Domain <String>] [-Server <String>]
```

Parameter	Type	Required	Description
Domain	String	No	-
Server	String	No	-

Returns: An object with Items, Success and ErrorMessage.

Get-AppPackages

Gets the packaged applications available for use in policy settings.

The Get-AppPackages cmdlet gets the packaged applications available in the specified domain, for use in policy settings that reference them. The result lists them in Items.

```
Get-AppPackages [-Domain <String>]
```

Parameter	Type	Required	Description
Domain	String	No	The DNS name of the Active Directory domain.

Returns: An object with Items, Success and ErrorMessage.

Get-DeviceInfo

Gets hardware device information for use in device-related policy settings.

The Get-DeviceInfo cmdlet gets hardware device information from a server, for use in device installation and other device-related policy settings. The result lists the devices in Items.

```
Get-DeviceInfo [-Server <String>] [-Domain <String>]
```

Parameter	Type	Required	Description
Server	String	No	(Optional) Server name to retrieve device information from.
Domain	String	No	The DNS name of the Active Directory domain.

Returns: An object with Items.

Sign-in and identity providers

Configure SAML and OpenID Connect sign-in, and SCIM provisioning. These change how people sign in to UPA: plan changes with your identity team and test them before production.

[Get-SamlConfig](#) · [Set-SamlConfig](#) · [Remove-SamlConfig](#) · [Set-InitialSamlConfig](#) · [Get-SamlMetadata](#) · [Get-OIDCConfig](#) · [Set-OIDCConfig](#) · [Remove-OIDCConfig](#) · [Get-ScimConfig](#) · [Set-ScimConfig](#) · [Remove-ScimConfig](#) · [Set-SCIMConnection](#) · [Sync-Scim](#) · [Get-ScimAuthToken](#) · [Get-ESMAuth](#) · [Get-ScimUser](#) · [Post-ScimUser](#) · [Put-ScimUser](#) · [Patch-ScimUser](#) · [Remove-ScimUser](#) · [Get-ScimGroup](#) · [Post-ScimGroup](#) · [Put-ScimGroup](#) · [Patch-ScimGroup](#) · [Remove-ScimGroup](#) · [New-ScimAddress](#) · [New-ScimEmailAddress](#) · [New-ScimOperation](#) · [New-ScimTypedValue](#)

Get-SamlConfig

Gets the list of SAML configurations.

The Get-SamlConfig cmdlet gets the list of SAML configurations. If a configuration name is specified, the cmdlet gets the details for the specified SAML configuration.

```
Get-SamlConfig [-Name <String>]
```

Parameter	Type	Required	Description
Name	String	No	(Optional) The name of the SAML configuration to retrieve. If no name is specified, the cmdlet returns a list of SAML configurations.

Set-SamlConfig

Creates or updates a SAML sign-in configuration.

The Set-SamlConfig sets configuration information for SAML authentication.

```
Set-SamlConfig [-Settings <SamlConfigSettings>] [-Xml <String>] [-File <String>]
[-Url <String>] [-IDP] [-ProvisioningMode <ProvisioningMode>] [-Name <String>]
[-ImportUsersFromScim] [-ImportGroupsFromScim] [-ScimConnectionName <String>]
[-NameIDFormat <SamlNameIdFormat>] [-MatchType <SamlConfigSettings.MatchType>]
[-MatchDomainName <String>] [-DomainUserProperty <String>] [-SIDProperty <String>]
[-EmailProperty <String>] [-DisplayNameProperty <String>]
[-UniqueIdProperty <String>] [-SignatureAlgorithm <SignatureAlgorithm>]
[-WantsRequestsSigned] [-ScimConnection <HAPIScimConfig>] [-LogoutUrl <String>]
[-IsDefault <Boolean>] [-OobExpireMinutes <Int>] [-OobShortIdLength <Int>]
[-OobShortIdChars <String>] [-PasscodeLength <Int>] [-PasscodeChars <String>]
```

Parameter	Type	Required	Description
Settings	SamlConfigSettings	No	The SAMLConfigSettings to save.
Xml	String	No	The SAML XML for the SAML IDP or SP to connect to.
File	String	No	The path to a file containing the SAML XML for the SAML IDP or SP to connect to.

Parameter	Type	Required	Description
Url	String	No	URL to download the SAML XML for the SAML IDP or SP to connect to.
IDP	Switch	No	If present, indicates that the SAML provider being configured is an Identity Provider
ProvisioningMode	ProvisioningMode	No	The Provisioning mode for the provider. Values: Manual, AutomaticProvisioning, ScimConnector, MatchToADAccounts, JustInTime
Name	String	No	Specify a name for this SAML configuration.
ImportUsersFromScim	Switch	No	Indicates whether to import users from SCIM.
ImportGroupsFromScim	Switch	No	Indicates whether to import groups from SCIM
ScimConnectionName	String	No	The name of the SCIM connection to use for import.
NameIDFormat	SamlNameIDFormat	No	The NameIDFormat that will be returned by the SAML provider. Values: Unspecified, EmailAddress, Entity, Transient, Persistent, Encrypted, X509SubjectName, WindowsDomainQualifiedName, Kerberos
MatchType	SamlConfigSettings.MatchType	No	If matching SAML accounts to AD accounts, indicates which property to use for matching. Values: NoMatch, UserName, SID, EmailAddress
MatchDomainName	String	No	If matching SAML accounts to AD accounts, indicates which AD domain to check for matches in.
DomainUserProperty	String	No	SAML Assertion property to map to DOMAIN\Username.
SIDProperty	String	No	SAML Assertion property to map to SID.
EmailProperty	String	No	SAML Assertion property to map to Email Address.
DisplayNameProperty	String	No	SAML Assertion property to map to Display Name.
UniqueldProperty	String	No	SAML Assertion property to map to Unique ID.
SignatureAlgorithm	SignatureAlgorithm	No	Signature algorithm (NoSignature,SHA_1,SHA_256) Values: NoSignature, SHA_1, SHA_256
WantsRequestsSigned	Switch	No	Force requests to be signed using SignatureAlgorithm.
ScimConnection	HAPIScimConfig	No	The SCIM connection to import SAML users from.
LogoutUrl	String	No	Logout URL for identity provider, if not using SAML SLO
IsDefault	Boolean	No	Set to false if this is not the default identity provider.
OOBExpireMinutes	Int	No	The time (in minutes) before an Out of Band login request expires
OOBShortIdLength	Int	No	The length of the request ID in the URI for Out of Band authentication

Parameter	Type	Required	Description
OOBShortIdChars	String	No	The set of characters used to generate the request URL for Out of Band authentication
PasscodeLength	Int	No	The length of the Passcode required to complete Out of Band authentication
PasscodeChars	String	No	The set of allowed character used to generate Passcodes for Out of Band authentication

Returns: DirectoryGeneralResponse

Remove-SamlConfig

Removes a SAML sign-in configuration.

The Remove-SamlConfig cmdlet removes a SAML configuration that was created with Set-SamlConfig.

```
Remove-SamlConfig -Name <String>
```

Parameter	Type	Required	Description
Name	String	Yes	The configuration name of the SAML configuration to remove.

Returns: DirectoryGeneralResponse

Set-InitialSamlConfig

Creates the config files the Gateway and Gatekeeper will use to connect to a 3rd party SAML\SCIM provider.

The Set-InitialSamlConfig cmdlet creates the config files the Gateway and Gatekeeper will use to connect to a 3rd party SAML\SCIM provider.

```
Set-InitialSamlConfig -OutFile <String> [-SamlXml <String>] [-SamlFile <String>]
  [-SamlUrl <String>] [-Name <String>] [-NameIDFormat <SamlNameIdFormat>]
  [-WantsRequestsSigned <Boolean>] [-SignatureAlgorithm <SignatureAlgorithm>]
  [-MatchType <SamlConfigSettings.MatchType>] [-MatchDomainName <String>]
  [-DomainUserProperty <String>] [-SIDProperty <String>] [-EmailProperty <String>]
  [-DisplayNameProperty <String>] [-UniqueIdProperty <String>]
  [-ScimConnection <HapiScimConnection>] [-RefreshInterval <Int>]
  [-AuthorizationUrl <String>] [-DefaultAdminUsers <String[]>]
  [-DefaultAdminGroups <String[]>] [-LogoutUrl <String>]
```

Parameter	Type	Required	Description
SamlXml	String	No	The SAML XML for the SAML IDP or SP to connect to.
SamlFile	String	No	The path to a file containing the SAML XML for the SAML IDP or SP to connect to.
SamlUrl	String	No	URL to download the SAML XML for the SAML IDP or SP to connect to.

Parameter	Type	Required	Description
Name	String	No	Specify a name for this SAML configuration.
NameIDFormat	SamlNameIDFormat	No	The name format that will be returned by the SAML IDP. Values: Unspecified, EmailAddress, Entity, Transient, Persistent, Encrypted, X509SubjectName, WindowsDomainQualifiedName, Kerberos
WantsRequestsSigned	Boolean	No	(Optional) Indicates whether to sign SAML requests. This setting overrides the setting in the metadata XML.
SignatureAlgorithm	SignatureAlgorithm	No	Specifies which algorithm to use when signing requests. This setting overrides the setting in the metadata XML. Values: NoSignature, SHA_1, SHA_256
MatchType	SamlConfigSettings.MatchType	No	If matching SAML accounts to AD accounts, indicates which property to use for matching. Values: NoMatch, UserName, SID, EmailAddress
MatchDomainName	String	No	If matching SAML accounts to AD accounts, indicates which AD domain to check for matches in.
DomainUserProperty	String	No	SAML Assertion property to map to DOMAIN\Username.
SIDProperty	String	No	SAML Assertion property to map to SID.
EmailProperty	String	No	SAML Assertion property to map to Email Address.
DisplayNameProperty	String	No	SAML Assertion property to map to Display Name.
UniqueIDProperty	String	No	SAML Assertion property to map to Unique ID.
OutFile	String	Yes	Path to file to save configuration to.
ScimConnection	HapiScimConnection	No	The SCIM connection to import SAML users from.
RefreshInterval	Int	No	The Refresh interval (in minutes) for the SCIM sync.
AuthorizationUrl	String	No	The authorization URL to get a SCIM token.
DefaultAdminUsers	String[]	No	The SCIM ids of users to set as default admins.
DefaultAdminGroups	String[]	No	The SCIM ids of groups to set as default admins.
LogoutUrl	String	No	Logout URL for identity provider, if not using SAML SLO

Get-SamlMetadata

Gets the UPA SAML metadata XML.

The Get-SamlMetadata cmdlet gets the SAML metadata XML for UPA. Use SP for the service provider metadata or IDP for the identity provider metadata.

```
Get-SamlMetadata -SP

Get-SamlMetadata -IDP
```

Parameter	Type	Required	Description
SP	Switch	Yes	Retrieve the UPA SAML Service Provider metadata XML.
IDP	Switch	Yes	Retrieve the UPA SAML Identity Provider metadata XML.

Returns: String

Get-OIDCConfig

Gets the list of OpenID Connect (OIDC) sign-in configurations.

The Get-OIDCConfig cmdlet gets the list of OpenID Connect (OIDC) configurations. If a configuration name is specified, the cmdlet gets the details for that configuration.

```
Get-OIDCConfig [-Name <String>]
```

Parameter	Type	Required	Description
Name	String	No	(Optional) The name of the OIDC configuration to retrieve. If no name is specified, the cmdlet returns a list of OIDC configurations.

Set-OIDCConfig

Creates or updates an OpenID Connect (OIDC) sign-in configuration.

The Set-OIDCConfig cmdlet sets configuration information for OpenID Connect (OIDC) sign-in.

```
Set-OIDCConfig [-Settings <SamlConfigSettings>] [-ConfigUrl <String>]
[-ClientId <String>] [-IdentityClaim <String>]
[-ProvisioningMode <ProvisioningMode>] [-Name <String>] [-ImportUsersFromScim]
[-ImportGroupsFromScim] [-ScimConnectionName <String>]
[-MatchType <SamlConfigSettings.MatchType>] [-MatchDomainName <String>]
[-DomainUserProperty <String>] [-SIDProperty <String>] [-EmailProperty <String>]
[-DisplayNameProperty <String>] [-UniqueIdProperty <String>]
[-SignatureAlgorithm <SignatureAlgorithm>] [-WantsRequestsSigned]
[-ScimConnection <HAPIScimConfig>] [-LogoutUrl <String>] [-IsDefault <Boolean>]
[-AdditionalProperties <String>] [-OobExpireMinutes <Int>] [-OobShortIdLength <Int>]
[-OobShortIdChars <String>] [-PasscodeLength <Int>] [-PasscodeChars <String>]
```

Parameter	Type	Required	Description
Settings	SamlConfigSettings	No	The configuration settings to save.
ConfigUrl	String	No	URL of the identity provider's OIDC configuration.
ClientId	String	No	The Client ID.
IdentityClaim	String	No	The name of the OIDC claim that contains the user's identity.

Parameter	Type	Required	Description
ProvisioningMode	ProvisioningMode	No	The Provisioning mode for the provider. Values: Manual, AutomaticProvisioning, ScimConnector, MatchToADAccounts, JustInTime
Name	String	No	Specify a name for this OIDC configuration.
ImportUsersFromScim	Switch	No	Indicates whether to import users from SCIM.
ImportGroupsFromScim	Switch	No	Indicates whether to import groups from SCIM
ScimConnectionName	String	No	The name of the SCIM connection to use for import.
MatchType	SamlConfigSettings.MatchType	No	If matching SAML accounts to AD accounts, indicates which property to use for matching. Values: NoMatch, UserName, SID, EmailAddress
MatchDomainName	String	No	If matching SAML accounts to AD accounts, indicates which AD domain to check for matches in.
DomainUserProperty	String	No	SAML Assertion property to map to DOMAIN\Username.
SIDProperty	String	No	SAML Assertion property to map to SID.
EmailProperty	String	No	SAML Assertion property to map to Email Address.
DisplayNameProperty	String	No	SAML Assertion property to map to Display Name.
UniqueldProperty	String	No	SAML Assertion property to map to Unique ID.
SignatureAlgorithm	SignatureAlgorithm	No	Signature algorithm (NoSignature,SHA_1,SHA_256) Values: NoSignature, SHA_1, SHA_256
WantsRequestsSigned	Switch	No	Force requests to be signed using SignatureAlgorithm.
ScimConnection	HAPIScimConfig	No	-
LogoutUrl	String	No	Logout URL for identity provider, if not using SAML SLO
IsDefault	Boolean	No	Set to false if this is not the default identity provider.
AdditionalProperties	String	No	Additional OIDC properties to send with the OIDC login request
OOBExpireMinutes	Int	No	The time (in minutes) before an Out of Band login request expires
OOBShortIdLength	Int	No	The length of the request ID in the URI for Out of Band authentication
OOBShortIdChars	String	No	The set of characters used to generate the request URL for Out of Band authentication
PasscodeLength	Int	No	The length of the Passcode required to complete Out of Band authentication
PasscodeChars	String	No	The set of allowed character used to generate Passcodes for Out of Band authentication

Returns: DirectoryGeneralResponse

Remove-OIDCConfig

Removes an OpenID Connect (OIDC) sign-in configuration.

The Remove-OIDCConfig cmdlet removes an OpenID Connect (OIDC) configuration that was created with Set-OIDCConfig.

```
Remove-OIDCConfig -Name <String>
```

Parameter	Type	Required	Description
Name	String	Yes	The configuration name of the OIDC configuration to remove.

Returns: DirectoryGeneralResponse

Get-ScimConfig

Gets the list of configured SCIM providers and their current status.

The Get-ScimConfig cmdlet gets the list of configured SCIM providers. If a Name is specified, the cmdlet retrieves the details about the specified SCIM configuration. If the -Status flag is specified, the cmdlet retrieves the status of the last SCIM SYNC for the specified provider.

```
Get-ScimConfig [-Name <String>] [-Status]
```

Parameter	Type	Required	Description
Name	String	No	(Optional) The name of the SCIM configuration to retrieve. If no name is specified, the cmdlet returns a list of SCIM configurations.
Status	Switch	No	Retrieves the status of the most recent SCIM Sync Job for the specified SCIM configuration.

Returns: String[], HAPIScimConfig, HAPIScimStatus

Set-ScimConfig

Creates or updates a SCIM configuration.

The Set-ScimConfig cmdlet creates a SCIM configuration, or updates an existing one, that provisions users and groups between a SCIM provider and Active Directory. To change an existing configuration, pass it from Get-ScimConfig with Config.

```
Set-ScimConfig -Name <String> [-Config <HAPIScimConfig>] [-Saml] [-RefreshMinutes <Int>]
[-LastUSN <Long>] [-Domain <String>] [-DomainController <String>] [-Server <String>]
[-BaseUrl <String>] [-HeaderName <String>] [-HeaderFormat <String>]
[-Token <String>] [-AuthorizationUrl <String>] [-NoCreateUsers] [-NoDeleteUsers]
[-NoUpdateUsers] [-NoCreateGroups] [-NoDeleteGroups] [-NoUpdateGroups] [-PullGroups]
[-UsePutForPatch] [-GroupPrefix <String>] [-GroupCreateDN <String>]
[-GroupLdapFilter <String>] [-UserLdapFilter <String>] [-UserJsonMap <String>]
[-MapExistingADUsers] [-ADMatchProperty <String>] [-SCIMMatchProperty <String>]
[-SamlNoUsers] [-SamlNoGroups]
```

Parameter	Type	Required	Description
Config	HAPIScimConfig	No	An existing HAPIScimConfig object, possibly returned from Get-ScimConfig.
Name	String	Yes	The name for this SCIM configuration.
Saml	Switch	No	-
RefreshMinutes	Int	No	Indicates how often UPA should perform the SCIM sync. If 0 is specified, the sync can be performed manually using the Sync-Scim Cmdlet.
LastUSN	Long	No	-
Domain	String	No	The AD domain for this SCIM connection.
DomainController	String	No	(Optional) The DNS name or IP address of the domain controller to query.
Server	String	No	The server name (and port) for the SCIM server (i.e. "https://scim.contoso.com:8443")
BaseUrl	String	No	The relative base URL for the SCIM server (i.e. "scim/v2")
HeaderName	String	No	The header name for the authorization token (default="Authorization")
HeaderFormat	String	No	The header format for the authorization token (default="Bearer {0}")
Token	String	No	The SCIM Authorization token to use.
AuthorizationUrl	String	No	A URL to query for an authorization token. This option is an alternative to specifying the Token parameter.
NoCreateUsers	Switch	No	Don't create SCIM users.
NoDeleteUsers	Switch	No	Don't delete SCIM users.
NoUpdateUsers	Switch	No	Don't update SCIM users.
NoCreateGroups	Switch	No	Don't create SCIM groups.
NoDeleteGroups	Switch	No	Don't delete SCIM groups.
NoUpdateGroups	Switch	No	Don't update SCIM groups.
PullGroups	Switch	No	Create AD groups for SCIM groups, if the AD group doesn't exist.
UsePutForPatch	Switch	No	Uses the SCIM PUT operation instead of PATCH to update group membership.
GroupPrefix	String	No	Prefix to append to AD groups, to avoid naming conflict with existing AD groups.
GroupCreateDN	String	No	Distinguished name of the OU to create AD groups in.
GroupLdapFilter	String	No	Used when creating/updating SCIM groups. Only AD groups matching the filter will be synced to SCIM
UserLdapFilter	String	No	Used when creating/updating SCIM users. Only AD users matching the filter will be synced to SCIM

Parameter	Type	Required	Description
UserJsonMap	String	No	Path to a JSON file containing a template SCIM user. To map a SCIM property to an AD property, specify the value of the SCIM property as "#ldapPropertyName"
MapExistingADUsers	Switch	No	Indicates whether to map SCIM users to existing AD users, using a specified property.
ADMatchProperty	String	No	When ADMatchProperty=true, the name of the AD property to match.
SCIMMatchProperty	String	No	When ADMatchProperty=true, the name of the SCIM property to match.
SamInoUsers	Switch	No	-
SamInoGroups	Switch	No	-

Returns: DirectoryGeneralResponse

Remove-ScimConfig

Removes a SCIM Configuration.

The Remove-ScimConfig cmdlet removes a SCIM configuration that was previously created using the Set-ScimConfig cmdlet.

```
Remove-ScimConfig -Name <String>
```

Parameter	Type	Required	Description
Name	String	Yes	The configuration name of the SCIM configuration to remove.

Returns: DirectoryGeneralResponse

Set-SCIMConnection

Sets Scim connection information.

The Set-SCIMConnection cmdlet saves the specified SCIM connection information in an environment variable, where it can be used by the other SCIM-related cmdlets.

```
Set-SCIMConnection -Server <String> -BaseUrl <String> [-Authorization <String>]
```

Parameter	Type	Required	Description
Server	String	Yes	URL of the SCIM server (i.e. https://servername).
BaseUrl	String	Yes	The Base URL for the SCIM API on the specified server (i.e. /SCIM/v2).
Authorization	String	No	An authorization bearer token to use for SCIM operations. Alias: auth.

Sync-Scim

Runs a synchronization for a SCIM configuration.

The Sync-Scim cmdlet runs a synchronization for the named SCIM configuration now, rather than waiting for its scheduled refresh.

```
Sync-Scim -Name <String> [-Database]
```

Parameter	Type	Required	Description
Name	String	Yes	-
Database	Switch	No	-

Get-ScimAuthToken

Generates an authorization token for a SCIM configuration.

The Get-ScimAuthToken cmdlet generates an authorization token for the named SCIM configuration.

```
Get-ScimAuthToken -ProviderName <String> [-DaysToExpiry <Int>]
```

Parameter	Type	Required	Description
ProviderName	String	Yes	The SCIM configuration name.
DaysToExpiry	Int	No	The number of days until the token expires. This cannot exceed the maximum token lifetime configured on the server.

Returns: String

Get-ESMAuth

Gets a SCIM authentication token from an ESM server.

The Get-ESMAuth cmdlet uses the specified URL to retrieve a SCIM authentication token from an ESM server.

```
Get-ESMAuth -Url <String>
```

Parameter	Type	Required	Description
Url	String	Yes	The URL (including username/password parameters) to get the authentication token.

Get-ScimUser

Gets User information from a SCIM provider.

The Get-ScimUser cmdlet gets a specified user, or performs a search to retrieve multiple users from a SCIM provider. If -ID is specified, retrieves the SCIM user with the specified ID.

```
Get-ScimUser [-Query <String>] [-Id <String>] [-Attributes <String[]>]  
[-SkipCount <Int>] [-MaxResults <Int>]
```

Parameter	Type	Required	Description
Query	String	No	A SCIM filter string used to limit the results returned.
Id	String	No	The unique ID of the user to retrieve.
Attributes	String[]	No	An array strings containing the property names of properties to return. Alias: atts.
SkipCount	Int	No	Indicates the number of items to skip. Alias: skip.
MaxResults	Int	No	Maximum number of items to return. Alias: max.

Returns: ScimUser[], ScimUser

Post-ScimUser

Creates a new SCIM user.

The Post-ScimUser cmdlet creates a new SCIM user.

```
Post-ScimUser -User <ScimUser>
```

```
Post-ScimUser -UserName <String> -Emails <ScimEmailAddress[]> [-ExternalId <String>]
  [-DisplayName <String>] [-GivenName <String>] [-FamilyName <String>]
  [-FormattedName <String>] [-NickName <String>] [-UserType <String>]
  [-Title <String>] [-Timezone <String>] [-PreferredLanguage <String>]
  [-Locale <String>] [-ProfileUrl <String>] [-Password <String>]
  [-Telephones <ScimTypedValue[]>] [-IMs <ScimTypedValue[]>]
  [-Photos <ScimTypedValue[]>] [-Addresses <ScimAddress[]>] [-Active <Boolean>]
```

Parameter	Type	Required	Description
User	ScimUser	Yes	A SCIMUser object containing the property values for the new user.
ExternalId	String	No	The ExternalID for the new user. If not specified, and external ID will be generated. Alias: exid.
UserName	String	Yes	The username for the new user. Alias: name.
DisplayName	String	No	The display name for the new user.
GivenName	String	No	The first name for the new user. Alias: first.
FamilyName	String	No	The surname for the new user. Alias: last.
FormattedName	String	No	Formatted name for the user (i.e. "Firstname I. Lastname").
NickName	String	No	-
UserType	String	No	-
Title	String	No	-
Timezone	String	No	-
PreferredLanguage	String	No	-
Locale	String	No	-

Parameter	Type	Required	Description
ProfileUrl	String	No	-
Emails	ScimEmailAddress[]	Yes	The email address for the new user. To add a user with multiple email addresses, specify a SCIMUser object containing the user's properties. Alias: email.
Password	String	No	The password for the new user. Alias: pswd.
Telephones	ScimTypedValue[]	No	-
IMs	ScimTypedValue[]	No	-
Photos	ScimTypedValue[]	No	-
Addresses	ScimAddress[]	No	-
Active	Boolean	No	-

Returns: ScimUser

Put-ScimUser

Updates the properties of a SCIM user.

The Put-ScimUser cmdlet updates the properties of a SCIM user.

```
Put-ScimUser -User <ScimUser>
```

Parameter	Type	Required	Description
User	ScimUser	Yes	A SCIM user object containing the new properties for the user. Accepts pipeline input.

Returns: ScimUser

Patch-ScimUser

Updates a SCIM user.

The Patch-ScimUser cmdlet updates a SCIM user by applying one or more patch operations created with New-ScimOperation.

```
Patch-ScimUser -User <ScimUser> -Ops <ScimOperation[]>
Patch-ScimUser -Id <String> -Ops <ScimOperation[]>
```

Parameter	Type	Required	Description
User	ScimUser	Yes	A SCIM user object containing the user to update. Accepts pipeline input.
Ops	ScimOperation[]	Yes	The patch operations to apply, created with New-ScimOperation. Alias: add.
Id	String	Yes	The Id of the SCIM user to update.

Returns: ScimUser

Remove-ScimUser

Removes a SCIM user.

The Remove-ScimUser cmdlet removes a SCIM user.

```
Remove-ScimUser -User <ScimUser>
```

```
Remove-ScimUser -Id <String>
```

Parameter	Type	Required	Description
User	ScimUser	Yes	A SCIMUser object representing the user to delete.
Id	String	Yes	The ID of the SCIM user to delete.

Returns: ScimUser

Get-ScimGroup

Gets Group information from a SCIM provider.

The Get-ScimGroup cmdlet gets a specified group, or performs a search to retrieve multiple groups from a SCIM provider. If -ID is specified, retrieves the SCIM group with the specified ID.

```
Get-ScimGroup [-Query <String>] [-Id <String>] [-Attributes <String[]>]  
[-SkipCount <Int>] [-MaxResults <Int>]
```

Parameter	Type	Required	Description
Query	String	No	A SCIM filter string used to limit the results returned.
Id	String	No	The unique ID of the group to retrieve.
Attributes	String[]	No	An array strings containing the property names of properties to return. Alias: atts.
SkipCount	Int	No	Indicates the number of items to skip. Alias: skip.
MaxResults	Int	No	Maximum number of items to return. Alias: max.

Returns: ScimGroup[], ScimGroup

Post-ScimGroup

Creates a new SCIM group.

The Post-ScimGroup cmdlet creates a new SCIM group.

```
Post-ScimGroup [-Group <ScimGroup>] [-ExternalId <String>] [-Name <String>]
```

Parameter	Type	Required	Description
Group	ScimGroup	No	A SCIM group object containing the properties of the group to create.

Parameter	Type	Required	Description
ExternalId	String	No	The External ID for the new SCIM group. If no External ID is specified, one will be generated. Alias: exid.
Name	String	No	The name for the new group.

Returns: ScimGroup

Put-ScimGroup

Updates a SCIM group.

The Put-ScimGroup cmdlet updates a SCIM group. Currently, only group name can be updated.

```
Put-ScimGroup -Group <ScimGroup>

Put-ScimGroup -Id <String> [-Name <String>] [-Members <ScimUser[]>]
```

Parameter	Type	Required	Description
Group	ScimGroup	Yes	A SCIMGroup object containing the properties of the group to update. Accepts pipeline input.
Id	String	Yes	The ID of the SCIM group to update.
Name	String	No	The new name for the SCIM group.
Members	ScimUser[]	No	An array of SCIMUser objects representing the new members of the group.

Returns: ScimGroup

Patch-ScimGroup

Updates a SCIM group.

The Patch-ScimGroup cmdlet updates a SCIM group. Currently, group membership is the only group attribute that can be updated.

```
Patch-ScimGroup -Group <ScimGroup>

Patch-ScimGroup -Id <String> [-AddMembers <ScimUser[]>] [-RemoveMembers <ScimUser[]>]
```

Parameter	Type	Required	Description
Group	ScimGroup	Yes	A SCIM group object containing the group to update. Accepts pipeline input.
Id	String	Yes	The Id of the SCIM group to update.
AddMembers	ScimUser[]	No	An array of SCIM users to add to the group. Alias: add.
RemoveMembers	ScimUser[]	No	An array of SCIM users to remove from the group. Alias: rmv.

Returns: ScimGroup

Remove-ScimGroup

Removes a SCIM group.

The Remove-ScimGroup cmdlet removes a SCIM group.

```
Remove-ScimGroup -Group <ScimGroup>
```

```
Remove-ScimGroup -Id <String>
```

Parameter	Type	Required	Description
Group	ScimGroup	Yes	The SCIMGroup object for the group to delete.
Id	String	Yes	The ID of the SCIM group to delete.

Returns: ScimGroup

New-ScimAddress

Creates a SCIM address object for use with Post-ScimUser and Put-ScimUser.

The New-ScimAddress cmdlet creates a SCIM address object for the addresses property of a SCIM user.

```
New-ScimAddress [-StreetAddress <String>] [-Locality <String>] [-Region <String>]  
[-Country <String>] [-PostalCode <String>] [-Formatted <String>] [-Type <String>]
```

Parameter	Type	Required	Description
StreetAddress	String	No	-
Locality	String	No	-
Region	String	No	-
Country	String	No	-
PostalCode	String	No	-
Formatted	String	No	-
Type	String	No	-

Returns: ScimAddress

New-ScimEmailAddress

Creates a SCIM email address object for use with Post-ScimUser and Put-ScimUser.

The New-ScimEmailAddress cmdlet creates a SCIM email address object for the emails property of a SCIM user.

```
New-ScimEmailAddress -Type <String> -Value <String> [-Primary <Boolean>]
```

Parameter	Type	Required	Description
Type	String	Yes	-
Value	String	Yes	-
Primary	Boolean	No	-

Returns: ScimEmailAddress

New-ScimOperation

Creates a SCIM patch operation for use with Patch-ScimUser and Patch-ScimGroup.

The New-ScimOperation cmdlet creates a SCIM patch operation - Add, Remove or Replace a value at a path - to pass to Patch-ScimUser or Patch-ScimGroup.

```
New-ScimOperation -Op <OperationType> -Path <String> -Value <Object>
```

Parameter	Type	Required	Description
Op	OperationType	Yes	Values: Add, Remove, Replace
Path	String	Yes	-
Value	Object	Yes	-

Returns: ScimOperation

New-ScimTypedValue

Creates a SCIM typed value, such as a phone number, for use with Post-ScimUser and Put-ScimUser.

The New-ScimTypedValue cmdlet creates a SCIM typed value, such as a phone number, for a SCIM user.

```
New-ScimTypedValue -Type <String> -Value <String>
```

Parameter	Type	Required	Description
Type	String	Yes	-
Value	String	Yes	-

Returns: ScimTypedValue

Agents

Manage agent computers and review agent usage.

[Get-AgentCounts](#) · [Remove-AgentComputer](#)

Get-AgentCounts

Gets Agent usage counts and details for the given time window.

The Get-AgentCounts cmdlet retrieves and calculates agent usage counts and details for the given time window.

```
Get-AgentCounts [-SkipDetails] [-StartDate <DateTime>] [-EndDate <DateTime>]
```

Parameter	Type	Required	Description
SkipDetails	Switch	No	-
StartDate	DateTime	No	Alias: start.
EndDate	DateTime	No	Alias: end.

Returns: AgentCountResponse

Remove-AgentComputer

Removes an agent.

The Remove-AgentComputer cmdlet removes an agent. To remove an agent, uninstall the agent from the machine. The agent uninstall will remove the agent computer, and it is not necessary to call this cmdlet. This cmdlet can be used to force the removal of an agent, when the machine is permanently offline.

```
Remove-AgentComputer -Guid <Guid> -Domain <String>
```

Parameter	Type	Required	Description
Guid	Guid	Yes	The GUID of the agent to remove.
Domain	String	Yes	The DNS name of the Active Directory domain.

Returns: DirectoryGeneralResponse

Appendix A — Reserved cmdlets

These cmdlets are installed with the tools but are used by UPA itself or reserved for future use. You do not need them to administer UPA.

Cmdlet	Purpose
New-AgentComputer	Adds a new agent computer.
Get-AgentPolicy	Gets the policy settings for a specified agent.
Get-AvailablePermissions	Gets the set of permissions that are supported by a UPA Provider.
Get-MaxPermissions	Gets the set of permissions that are supported by a UPA Provider.
Copy-GPOFiles	Copies a GPO's files from SYSVOL, and optionally reformats preferences XML files.

Cmdlet	Purpose
Get-SamlResponseFormat	Gets the custom SAML response format, if one has been specified using the Set-SamlResponseFormat cmdlet.
Set-SamlResponseFormat	Sets the custom SAML response format.

Appendix B — Values for longer enumerations

A parameter of one of these types takes one of the values listed. Where the parameter type ends in `[]`, it takes several.

DeploymentAssigneeType

GPO · SCCMBaseline · SCCMAntiMalware · Agent · AllLinuxAgents · AllMacAgents · AllWindowsAgents · AllAgents · IntuneDeviceConfig · O365 · RepGP · RepCloudOU · AllEDRAgents

FilterOperation

None · GreaterThan · GreaterThanOrEqual · Equals · LessThan · LessThanOrEqual · NotEquals · Contains · NotContains · StartsWith · EndsWith · Like

GPPolicyProviderType

Password · Scripts · Security · Registry · LinuxSecurity · MACSecurity · SCCMBaseline · SCCMAntiMalware · Intune · Office365 · Citrix · EDR · Preferences · WindowsSecurity

MsftGPPermissionType

None · GpoApply · GpoRead · GpoEdit · GpoEditDeleteModifySecurity · GpoCustom · WmiFilterEdit · WmiFilterFullControl · WmiFilterCustom · StarterGpoRead · StarterGpoEdit · StarterGpoFullControl · StarterGpoCustom · SomCreateWmiFilter · SomWmiFilterFullControl · SomCreateGpo · SomCreateStarterGpo · SomLogging · SomPlanning · SomLink

Permissions

None · CreateNewDomain · DeleteDomain · EditDomainMaps · ViewOU · CreateOU · DeleteOU · RenameOU · ViewUP · Migrate · ApproveUP · SubmitUPForApproval · CreateUP · DeleteUP · ModifyUP · ModifyUPLinks · UndoCheckout · RenameUP · Rollback · Audit · FullOUControl · FullUPControl · ViewAD · ModifyAD · DeleteAD · FullADControl · ViewGPO · ExportToAD · ImportFromAD · ModifyGPO · DeleteGPO · ViewDelegation · FullDelegationControl · ModifyDelegation · ViewAgent · AddAgent · RemoveAgent · FullAgentControl · CreateGPO · FullDomainControl · FullGPOControl · FullControl

WindowsGroupType

Unknown · CreatedBySystem · Global · DomainLocal · Universal · Azman_AppBasic · Azman_AppQuery · SecurityGroup · GlobalSecurity · DomainLocalSecurity · UniversalSecurityGroup

Getting help

For help with UPA PowerShell Tools, contact your Full Armor representative or email support@fullarmor.com.