

# PowerShell Quick Reference - Security / Compliance Centers

(v1.03)  
2022/10/05

Connecting to Security and Compliance Center (SCC)	Listing Cmdlets for the SCC	Cmdlet Changes since 2017
Install-Module ExchangeOnlineManagement Import-Module ExchangeOnlineManagement Connect-IPSSession -UserPrincipalName damian@practicalpowershell.com	<b>List all Commands for the Security and Compliance Center</b> \$Name = (Get-Module   where {\$_.ModuleType -eq 'Script'}).Name Get-Command   Where {\$_.ModuleName -eq \$Name}	<b>Security and Compliance Center</b> 12.31.2017 158 cmdlets 10.05.2022 333 cmdlets

eDiscovery Admin
<p><b>eDiscovery Admin</b> - eDiscovery Admins create searches/holds on mailboxes, SharePoint Sites and OneDrive locations. They also manage/create eDiscovery case, content searches and add members to handle these cases.</p> <p><b>List current eDiscovery Admins – There are zero in a greenfield Office 365 Tenant</b></p> Get-eDiscoveryCaseAdmin <b>New eDiscovery Case Admin</b> Add-eDiscoveryCaseAdmin -User damian@practicalpowershell.com <b>Remove an eDiscovery Admin</b> Remove-eDiscoveryCaseAdmin -User damian@practicalpowershell.com <b>Replace Current eDiscovery Admin</b> Update-eDiscoveryCaseAdmin -Users john@domain.com,jane@domain.com

AutoLabelling (New)
<p><b>List existing Auto Labelling Policies and Rules</b></p> Get-AutoSensitivityLabelPolicy <b>or</b> Get-AutoSensitivityLabelPolicy 'Auto Label Marketing' Get-AutoSensitivityLabelRule <b>or</b> Get-AutoSensitivityLabelRule 'Bank Account Numbers'
<p><b>Create new Auto Labelling Policies and Rules</b></p> New-AutoSensitivityLabelPolicy -Name "AL-BankAccoutPolicy" -ExchangeLocation All -Mode TestWithoutNotifications -ApplySensitivityLabel "Bank Account Numbers"
<p>New-AutoSensitivityLabelRule -Name "AL-BankAcctRule" -Policy "AL-BankAccoutPolicy" -ContentContainsSensitiveInformation @"{name}="U.S. Bank Account Number";"mincount="1"} -Workload Exchange</p>
<p><b>Remove existing Auto Labelling Policies and Rules</b></p> Remove-AutoSensitivityLabelPolicy "AL-BankAccoutPolicy" Remove-AutoSensitivityLabelRule "AL-BankAcctRule"
<p><b>Change existing Auto Labelling Policies and Rules</b></p> Set-AutoSensitivityLabelPolicy "AL-BankAccoutPolicy" -AddSharePointLocation 'my URL' Set-AutoSensitivityLabelRule "AL-BankAcctRule" -Comment 'Bank Acc # Autolabel Rule'

Get-Help
<p><b>Getting Help</b></p> Get-Help <command> Get-Help <command> -Examples Get-Help <command> -Full <b>Examples</b> Get-Help Set-ComplianceTag Get-Help Set-ComplianceTag -Examples Get-Help Set-ComplianceTag -Full

Application Compliance Policy (SCC)
Get-AppRetentionCompliancePolicy Get-AppRetentionComplianceRule New-AppRetentionCompliancePolicy New-AppRetentionComplianceRule Remove-AppRetentionCompliancePolicy Remove-AppRetentionComplianceRule Set-AppRetentionCompliancePolicy Set-AppRetentionComplianceRule

**Documentation:** <https://docs.microsoft.com/en-us/powershell/exchange/office-365-scc/office-365-scc-powershell>  
**Microsoft Defender (formerly Security) Admin:** <https://security.microsoft.com> / **Purview Compliance Admin:** <https://compliance.microsoft.com>

Role Groups in the SCC
<p><b>Role Group Cmdlets:</b></p> Get-RoleGroup – User 'Get-RoleGroup   FL' to get a detailed list of accounts in the SCC New-RoleGroup – Add a custom group, with specific roles in the SCC Remove-RoleGroup – Remove only custom and not built-in Role Groups Set-RoleGroup – Modify settings on existing Role Groups <b>Cmdlet Usage:</b> Get-RoleGroup   Where {\$_.Name -like '*admin*'}   Ft New-RoleGroup 'View-Only Auditor' -Roles 'View-Only Audit Logs' -Members George Remove-RoleGroup -Name 'View-Only Auditor' Set-RoleGroup -Name 'View-Only Auditor' -Description "Users with View Only Auditing" \$CSV = Import-CSV "CustomGroupDescriptions.csv" Foreach (\$Group in \$CSV) { Set-RoleGroup -Name \$Group.Name -Description \$Group.Description }

DLP Sensitive Information Types
<p><b>Find existing Sensitive Information Types:</b></p> Get-DlpSensitiveInformationType <b>Create new Sensitive Information Type with Fingerprints:</b> \$Content01 = Get-Content "\\File01\HR\EmployeeInfo.docx" -Encoding byte \$FingerPrint01 = New-DlpFingerprint -FileData \$Content01 -Description "Confidential Employee Information" New-DlpSensitiveInformationType -Name "Confidential Employee Information" -Fingerprints \$FingerPrint01 -Description "Sensitive Employee Information - HR" <b>Remove old unused Sensitive Information Types:</b> Remove-DlpSensitiveInformationType – Name "Confidential Employee Information" <b>Change an existing Sensitive Information Type:</b> Set-DlpSensitiveInformationType – Name "Confidential Employee Information"

# PowerShell Quick Reference - Security / Compliance Centers

(v1.03)  
2022/10/05

## DLP CMDLETS

Get-DlpCompliancePolicy  
Get-DlpComplianceRule  
Get-DlpComplianceRuleV2  
Get-DlpDetectionsReport  
Get-DlpKeywordDictionary  
Get-DlpSensitiveInformationType  
Get-DlpSensitiveInformationTypeRulePackage  
Get-DlpSiDetectionsReport  
Migrate-DlpFingerprint  
New-DlpCompliancePolicy  
New-DlpComplianceRule  
New-DlpComplianceRuleV2  
New-DlpFingerprint  
New-DlpKeywordDictionary  
New-DlpSensitiveInformationType  
New-DlpSensitiveInformationTypeRulePackage  
Remove-DlpCompliancePolicy  
Remove-DlpComplianceRule  
Remove-DlpComplianceRuleV2  
Remove-DlpKeywordDictionary  
Remove-DlpSensitiveInformationType  
Remove-DlpSensitiveInformationTypeRulePackage  
Set-DlpCompliancePolicy  
Set-DlpComplianceRule  
Set-DlpComplianceRuleV2  
Set-DlpKeywordDictionary  
Set-DlpSensitiveInformationType  
Set-DlpSensitiveInformationTypeRulePackage

## Cmdlet Highlight

**Get-SCInsights** – provides user totals per workloads – ExO, Archive, SharePoint, OneDrive and more

## Device Compliance

**To use Device Management cmdlets – Enable MDM for tenant first:**  
<https://support.office.com/en-us/article/overview-of-mobile-device-management-mdm-for-office-365-faa7d8e5-645d-4d59-839c-c8d4c1869e4a>

**New Device Rule – Tenant Wide, Less Options**  
New-DeviceTenantRule

**New Device Rule – Very Specific Configuration, More Options**  
New-DeviceConfigurationRule

**\*\* Note** the two cmdlet above have Set, Get and Remove Verbs as well

**Device Rules can be used in conjunction with Conditional Access**

Get-DeviceConditionalAccessPolicy  
Get-DeviceConditionalAccessRule  
New-DeviceConditionalAccessPolicy  
New-DeviceConditionalAccessRule  
Remove-DeviceConditionalAccessPolicy  
Remove-DeviceConditionalAccessRule  
Set-DeviceConditionalAccessPolicy  
Set-DeviceConditionalAccessRule

## REGEX Testing / Reference

### RegEx Testing

<https://regex101.com/>  
<https://regexr.com/>  
<http://osherove.com/tools>

### Microsoft RegEx Reference

<https://docs.microsoft.com/en-us/dotnet/standard/base-types/regular-expression-language-quick-reference>

## Future Cmdlets (Currently Not Working)

Get-LongTermAuditItems                      Get-InformationBarrierReportDetails  
Get-LongTermAuditStats                      Get-InformationBarrierReportSummary

## Created By:

**Damian Scoles**  
**Microsoft MVP**  
Book Author  
[practicalpowershell.com](http://practicalpowershell.com)  
[powershellgeek.com](http://powershellgeek.com)  
[m365securitybook.com](http://m365securitybook.com)  
[@PPowerShell](https://twitter.com/PPowerShell)

## Helpful Tips

Tab through parameters to see all available  
Check for latest module version  
Read the latest Microsoft Docs for SCC  
Read Teams MVP blogs for more tips  
Use MFA for better security  
Need Help – ‘Get-Help’  
Read cmdlet Synopsis for functionality

## Reporting Cmdlets

Get-DataRetentionReport  
Get-DeviceComplianceDetailsReport  
Get-DeviceComplianceDetailsReportFilter  
Get-DeviceComplianceReportDate  
Get-DeviceComplianceSummaryReport  
Get-DeviceComplianceUserReport  
Get-DlpDetectionsReport  
Get-DlpSiDetectionsReport  
Get-MailFilterListReport  
Get-SupervisoryReviewPolicyReport  
Get-SupervisoryReviewReport

## More On PowerShell

**Windows PowerShell Blog**  
[blogs.msdn.com/b/powershell](http://blogs.msdn.com/b/powershell)  
**Script Center**  
[technet.microsoft.com/scriptcenter](http://technet.microsoft.com/scriptcenter)  
**PowerShell Tips of the Week**  
[www.practicalpowershell.com/blog](http://www.practicalpowershell.com/blog)  
**PowerShell Team – GitHub**  
<https://github.com/powershell>

## DLP Fingerprinting

```
$RDDoc = Get-Content "Z:\RD\ResearchDoc-Contoso.docx" -Encoding byte
$RDDocFingerPrint = New-DlpFingerprint -FileData $RDDoc -Description "Research and Development Doc"
New-DlpSensitiveInformationType -Name "RD Document Fingerprint" -Fingerprints $RDDoc1FingerPrint -Description "Research and Development Doc - CONFIDENTIAL."
```

# PowerShell Quick Reference - Security / Compliance Centers

(v1.03)

2022/10/05

## Working with Role Groups

### Add User to Role Group

```
Add-RoleGroupMember -Identity Reviewer -Member Damian  
Add-RoleGroupMember -Identity ComplianceAdministrator -Member "John Smith"  
Add-RoleGroupMember -Identity eDiscoveryManager -Member "Scott Schnoll"
```

### Verify Users in Role Group

```
Get-RoleGroupMember -Identity Reviewer  
Get-RoleGroupMember -Identity ComplianceAdministrator  
Get-RoleGroupMember -Identity eDiscoveryManager
```

### Remove Users from Role Group

```
Remove-RoleGroupMember -IdentityReviewer -Member "Greg Taylor"  
Remove-RoleGroupMember -Identity ComplianceAdministrator -Member "Van Hybrid"  
Remove-RoleGroupMember -Identity eDiscoveryManager -Member "Jason Sherry"
```

### Update Role Group MemberShip

```
Update-RoleGroupMember -Identity Reviewer -Members "Damian","Dave"
```

## Working with Compliance Cases

### Create New Case

```
New-ComplianceCase -Name "Case # 430" -Description "Legal Case-R&D-05-2021"
```

### Add Compliance Case Members

```
Add-ComplianceCaseMember -Case "Case # 430" -Member damian@practicalpowershell.com  
Add-ComplianceCaseMember -Case "Case # 430" -Member dave@practicalpowershell.com
```

### Add Searches and Holds to the Case

```
New-CaseHoldPolicy -Name "Hold - Damian" -Case "Case # 430" -ExchangeLocation "John"  
New-ComplianceSearch -Name "Secret Meetings" -ExchangeLocation Damian -  
ContentMatchQuery "subject:Secret Meetings"
```

### Start the Search and apply a Search Action

```
Start-ComplianceSearch -Identity "Secret Meetings"  
New-ComplianceSearchAction -SearchName "Secret Meetings" -Export
```

### View Existing Compliance Cases

```
Get-ComplianceCase
```

**Security and Compliance Center PowerShell Book:** <https://www.practicalpowershell.com>

## Compliance Holds and Tags

### Create a new compliance tag:

```
New-ComplianceTag -Name "R&D" -RetentionAction Delete -RetentionDuration 365 -  
RetentionType TaggedAgeInDays
```

### List all current Compliance Tags

```
Get-ComplianceTag
```

### Removing and existing Compliance Tag

```
Remove-ComplianceTag -Name "R&D"
```

### Modifying an existing tag by adding a reviewer

```
Set-ComplianceTag -Name "R&D" -Reviewer damian@practicapowerhsell.com
```

### First, create a Hold Compliance Policy

```
New-HoldCompliancePolicy -Name "Case 5412-10" -ExchangeLocation john@standard.net
```

### Then create one or more Hold Compliance Rules

```
New-HoldComplianceRule -Policy "Case 5412-10" -Name "Hold 2021" -ContentDateFrom "06/  
01/2021" -ContentDateTo "6/30/21"
```

### Removing policies or rules

```
Remove-HoldCompliancePolicy "Case 5412-10"  
Remove-HoldComplianceRule "Hold 2021"
```

### Modify existing rules or policies:

```
Set-HoldCompliancePolicy -Name "Case 5412-10" -SharePointLocation "http://  
standard.sharepoint.com/sites/Teams/R&D"  
Set-HoldComplianceRule -Name "Hold 2021" -ContentDateFrom "07/01/21"
```

### List policies or rules that were created previously

```
Get-HoldCompliancePolicy  
Get-HoldComplianceRule -Name "Hold 2017"
```

## Security, Privacy and Compliance Blog

<https://techcommunity.microsoft.com/t5/Security-Privacy-and-Compliance/bg-p/securityprivacycompliance>

## Permissions in Security and Compliance Center

<https://docs.microsoft.com/en-us/office365/securitycompliance/permissions-in-the-security-and-compliance-center>

# PowerShell Quick Reference - Security / Compliance Centers

(v1.03)  
2022/10/05

4

## Admin Audit Log

### View Default Admin Audit Log Settings

Get-AdminAuditLogConfig

### Search the Admin Audit Log and send Email of results

New-AdminAuditLogSearch -StartDate 8/1/18 -EndDate 8/15/18 -StatusMailRecipients damian@practicalpowershell.com

### Disable/Enable Office 365 Admin Audit logs

Set-AdminAuditLogConfig -UnifiedAuditLogIngestionEnabled \$False

Set-AdminAuditLogConfig -UnifiedAuditLogIngestionEnabled \$True

*\*\* Note – Changes (using Set) need to be performed in Exchange Online PowerShell*

### New Unified Log Search – Exchange, SharePoint, OneDrive, Intune, AzureAD and more!

Search-UnifiedAuditLog -StartDate 10/1/2018 -EndDate 10/24/18

**Or SharePoint Only** - Search-UnifiedAuditLog -StartDate 10/1/2018 -EndDate 10/24/18 -RecordType SharePoint

## Auditing

### Change Audit Config

Set-AuditConfig -Workload Exchange,SharePoint,OneDriveForBusiness,Intune

### Audit all operations for a workload:

New-AuditConfigurationPolicy -Workload SharePoint

### Remove existing Audit Configuration Policy

Remove-AuditConfigurationPolicy 91f20f6f-7ef9-4561-9a38-d771452d5e45

### Audit specific operations in a workload

New-AuditConfigurationRule -Workload Exchange,SharePoint -AuditOperation Delete

### Modify existing Audit Configuration Rule

Set-AuditConfigurationRule

### Remove existing Audit Configuration Rule

New-AuditConfigurationRule -Identity <GUID of Rule>

### Current Configuration:

Get-AuditConfig

Get-AuditConfigurationPolicy

Get-AuditConfigurationRule

**Create Custom XML for DLP:** <http://www.powershellgeek.com/adventures-in-custom-dlp-rules-part-one/>

## DLP Keyword Dictionary

Create a list of keywords to be used by DLP to protect information in your tenant

### Check settings on Existing Dictionary:

Get-DlpKeywordDictionary -Name "Technical Docs"

### Create New DLP Keywords Dictionary

\$DLPKeywords = "Technical Specifications, Research Grant, Development Methodologies"

\$EncodedDLPKeywords = [system.Text.Encoding]::UTF8.GetBytes(\$DLPKeywords);

New-DlpKeywordDictionary -Name "Technical Docs" -Description "Keywords appearing in internal docs" -FileData \$EncodedDLPKeywords

### Remove an unneeded dictionary

Remove-DlpKeywordDictionary -Name "Technical Docs"

### Modify an Existing Dictionary (removing keywords in this case)

\$DLPKeywords = "Technical Specifications, Development Methodologies"

\$EncodedDLPKeywords = [system.Text.Encoding]::UTF8.GetBytes(\$DLPKeywords);

Set-DlpKeywordDictionary -Name "Technical Docs" -FileData \$EncodedDLPKeywords

## Supervisory Review

### First we need to create a Supervisory Policy as none exist by default:

New-SupervisoryReviewPolicyV2 -Name "R&D" -Reviewers george@cooltoys.com -Comment "Monitory R&D emails"

### Then create one or more Supervisory Rules:

New-SupervisoryReviewRule -SamplingRate 50 -Policy "R&D" -Condition (Reviewee:damian@cooltoys.com)

### Grab reports or information on the rules / policies created:

Get-SupervisoryReviewPolicyReport, Get-SupervisoryReviewPolicyV2

Get-SupervisoryReviewReport, Get-SupervisoryReviewRule

### Remove a policy (\*\* No cmdlet for removing a rule):

Remove-SupervisoryReviewPolicyV2

### Modify existing rules/policies

Set-SupervisoryReviewPolicyV2 -Name "R&D" -Reviewers "greg@cooltoys.com"

Set-SupervisoryReviewRule -SamplingRate 25 -Policy "R&D"

## Information Barriers

Information Barriers are a logical construct that prevents communication between groups of people. Any of the people that are blocked from communicating need to be synced to Azure AD. The filters for users are based off of Azure AD users and the attributes that are allowed for filters.

### Create a new Information Barrier Policy:

```
New-InformationBarrierPolicy -Name 'HR-Research' -AssignedSegment HR -SegmentsBlocked Research -State InActive
```

### List all Information Barrier Policies:

```
Get-InformationBarrierPolicy | Ft
```

### Remove an existing Information Barrier Policy:

```
Remove-InformationBarrierPolicy
```

### Change settings on existing Information Barrier Policy:

```
Set-InformationBarrierPolicy
```

### Kick off process to segment accounts:

```
Start-InformationBarrierPoliciesApplication
```

### Stop the process of segmenting accounts

```
Stop-InformationBarrierPoliciesApplication
```

### Check on the process of this application:

```
Get-InformationBarrierPoliciesApplicationStatus
```

### Verify a policy is applies to a user:

```
Get-InformationBarrierRecipientStatus -Identity JohnSmith
```

### Validate Information Barrier Policies:

```
Test-InformationBarrierPolicy
```

## Unified Audit Log Retention: <https://docs.microsoft.com/en-us/microsoft-365/compliance/audit-log-retention-policies>

Unified Audit Log Retention Policies determine how to handle audit logs for a tenant:

### List the settings of a Policy:

```
Get-UnifiedAuditLogRetentionPolicy
```

### Create a new Policy:

```
New-UnifiedAuditLogRetentionPolicy -Name "SharePoint Audit Policy" -Description "Six month retentionpolicy SharePoint log items" -RecordTypes SharePoint -RetentionDuration SixMonths -Priority 1
```

### Remove and Existing Policy:

```
Remove-UnifiedAuditLogRetentionPolicy
```

### Change Settings on an existing Policy:

```
Set-UnifiedAuditLogRetentionPolicy "SharePoint Audit Policy" -Priority 100
```

### Change record types for an existing Policy:

```
Set-UnifiedAuditLogRetentionPolicy "Office 365 Audit Policy" -RecordTypes SharePoint, ExchangeAdmin, MicrosoftTeams, Yammer, Sway
```

## Insider Risk

Insider risk policies are designed to help identify inadvertent and/or suspicious internal activity. Note that these are very new cmdlets and help is hard to find.

### Create a new Insider Risk Policy:

```
New-InsiderRiskPolicy -Name CheckForLeaks -InsiderRiskScenario LeakOfInformation
```

### List all existing Insider Risk Policies:

```
Get-InsiderRiskPolicy
```

### List one Insider Risk Policy

```
Get-InsiderRiskPolicy CheckForLeaks
```

### Remove an existing Insider Risk Policy

```
Remove-InsiderRiskPolicy CheckForLeaks
```

### Change settings on an existing Insider Risk Policy:

```
Set-InsiderRiskPolicy CheckForLeaks -AddExchangeLocation Brian
```

### Disable an existing Insider Risk Policy

```
Set-InsiderRiskPolicy test -Enabled $False
```

\*\* Requires E5 or E3 with Microsoft E5 Compliance add-on

## Quarantine

### Remove Quarantine Messages:

```
Get-QuarantineMessage | Delete-QuarantineMessage
```

```
$ID = (Get-QuarantineMessage | Where {$_.Type -eq 'High Confidence Phish'}).Identity
```

```
Delete-QuarantineMessage -identity $ID
```

### Export quarantined email for review (locate message and export to txt file:

```
$ID = (Get-QuarantineMessage | Where {$_.Type -eq 'High Confidence Phish'}).Identity
```

```
$ExportMessage = Export-QuarantineMessage -Identity $ID
```

```
$Encoding = [Convert]::FromBase64String($ExportMessage.Eml)
```

```
[IO.File]::WriteAllBytes("C:\scripts\Export1.txt", $Encoding)
```

### Find Quarantine messages for 2020:

```
Get-QuarantineMessage -StartReceivedDate 01/01/2020 -EndReceivedDate 12/30/2020
```

### Find quarantine messages not reported as false positives:

```
Get-QuarantineMessage -Reported $False | Ft -Auto
```

### Get an email header, using the message identity stored in \$ID:

```
Get-QuarantineMessageHeader $ID
```

### Preview a Quarantined message using the same \$ID variable as before:

```
Preview-QuarantineMessage $ID
```

### Release a message for an end user:

```
Release-QuarantineMessage <Message Identity>
```

## File Plans

File Plan Manager is a new feature that Microsoft introduced in 2019 to the Security and Compliance Center. The intended purpose is to help apply tags to data in your tenant to help search for and discover. All Get-\* cmdlets output in a list format by default.

### Special Cmdlets for File Plans:

#### **Export an existing File Plan Property:**

Export-FilePlanProperty

#### **Import a File Plan Property (using CSV file)**

Import-FilePlanProperty -RawCSV 'FilePlanProperty1.csv'

#### **List the structure of the File plan:**

Get-FilePlanPropertyStructure

#### **Retrieve the Json for the structure:**

Get-FilePlanPropertyStructure | Fl FilePlanStructureJSON

### Property - Authorities

#### **List any existing Authorities:**

Get-FilePlanPropertyAuthority | Ft -Auto

#### **Create a new Authority:**

New-FilePlanPropertyAuthority -Name 'HR'

#### **Remove an existing Authority:**

Remove-FilePlanPropertyAuthority HR

#### **Change a setting on an existing Authority:**

Set-FilePlanPropertyAuthority 'IT' -DisplayName 'Information Technology'

### Property Categories

#### **List any existing Categories:**

Get-FilePlanPropertyCategory | Ft -Auto

#### **Create a new Category:**

New-FilePlanPropertyCategory -Name 'Internal Review'

#### **Remove an existing Category:**

Remove-FilePlanPropertyCategory 'Internal Review'

#### **Change settings on an existing category:**

Set-FilePlanPropertyCategory 'Internal Review' -Disabled \$True

## Labels – Add tags from File Plan

Adding File Plan information can be done with PowerShell. It is NOT straightforward. There are no parameters to handle this. There are two ways to add any of these properties to an existing or new label – either use the Security and Compliance Center and the GUI interface or use PowerShell. PowerShell takes a bit of work. A working example is included in my '*Practical PowerShell Security and Compliance Center*' book due to the complexity. There isn't sufficient space in a Quick Reference for breaking this out.

**Full Security and Compliance Center PowerShell Book:**

<https://PracticalPowershell.com>

### Property - Citation

#### **List any existing Citations configured:**

Get-FilePlanPropertyCitation | Ft -Auto

#### **Create new Citation (FTC – Truth in Advertising example):**

\$CitationURL = 'https://www.ftc.gov/news-events/media-resources/truth-advertising'

\$Name = 'Truth in Advertising'

\$CitationJurisdiction = 'Federal Trade Commission (FTC)'

New-FilePlanPropertyCitation -Name \$Name -CitationURL \$CitationURL -

CitationJurisdiction \$CitationJurisdiction

#### **Remove an existing Citation:**

Remove-FilePlanPropertyCitation 'Truth in Advertising'

#### **Change a setting on an existing Citation:**

Set-FilePlanPropertyCitation \$Name -DisplayName 'Truth in Advertising (2019)'

### Property - Department

#### **List any existing Department:**

Get-FilePlanPropertyDepartment | Ft -Auto

#### **Create a new Department:**

New-FilePlanPropertyDepartment

Remove an Existing Department

Remove-FilePlanPropertyDepartment

Change a setting on an existing Department:

Set-FilePlanPropertyDepartment

### Property – ReferenceID

#### **List any existing ReferenceID:**

Get-FilePlanPropertyReferenceId | Ft -Auto

#### **Create a new ReferenceID:**

New-FilePlanPropertyReferenceId -Name 'ID 821'

#### **Remove an existing ReferenceID:**

Remove-FilePlanPropertyReferenceId 'ID 812'

#### **Change the settings on an existing ReferenceID:**

Set-FilePlanPropertyReferenceId 'ID 812' -Disabled \$False -Comment 'Reinstated 12/12/19'

### Property SubCategory (Requires a parent Category to exist)

#### **List any existing SubCategory**

Get-FilePlanPropertySubCategory | Ft -Auto

#### **Create new SubCategory:**

New-FilePlanPropertySubCategory -Name 'Litigant - Employee 4569' -ParentId '73293e6a-5612-459e-9b74-fc9022d9e2aa'

#### **Remove an existing SubCategory"**

Remove-FilePlanPropertySubCategory -Name 'Litigant - Employee 4569'

#### **Change settings on an existing Subcategory:**

Set-FilePlanPropertySubCategory 'Litigant - Employee 4569' -Disabled \$True -Comment 'No longer needed – 01/03/2020'

## Exact Data Match

### List already created Exact Data Match schema

```
Get-DlpEdmSchema  
Get-DlpEdmSchema -Identity 'Bank Customer Records'
```

### Create New Exact Data Match schema

```
$EDMSchema = Get-Content .\EmployeeRecord.xml -Encoding Byte -ReadCount 0  
New-DlpEdmSchema -FileData $EDMSchema -Confirm:$true
```

### Delete an existing Exact Data Match schema

```
Remove-DlpEdmSchema  
Remove-DlpEdmSchema -Identity 'Bank Customer Records'
```

### Modiy an existing Exact Data Match schema

```
Set-DlpEdmSchema  
$EDMSchema2 = Get-Content .\EmployeeRecord-2022.xml -Encoding Byte -ReadCount 0  
New-DlpEdmSchema -FileData $EDMSchema2 -Confirm:$true
```

Script to Update EDM schemas for data refreshes:

EXAMPLE

## Sensitivity Labels

### List existing labels and label policies

```
Get-Label or Get-Label | Where {$_.ContentType -eq 'PurviewAssets'}  
Get-LabelPolicy or Get-LabelPolicy | Where {$_.ModernGroupLocation -ne $Null} | ft name
```

### Create New Labels and Public Labels (new policy)

```
New-Label -DisplayName 'Confidential' -Name 'Confidential' -ToolTip 'Content is  
Confidential and for Internal User ONLY!'  
New-LabelPolicy -Name 'Confidential Label' -Labels 'Confidential'
```

### Remove Labels or Label Policies

```
Remove-Label 'Confidential'  
Remove-LabelPolicy 'Confidential Label'
```

### Change settings on Labels or Label Policies

```
Set-Label -Identity "Ultra Confidential" -EncryptionRightsDefinitions  
"quest.com:VIEW,VIEWRIGHTSDATA,DOCEDIT,PRINT"  
Set-LabelPolicy -Identity "General Sensitivity Policy" -AdvancedSettings  
@{EnableContainerSupport="True"}
```

## Phish Sim Override

### List existing Phish Sim Policies or Rules

```
Get-PhishSimOverridePolicy  
Get-PhishSimOverrideRule
```

### Create New Phish Sim Override Policies and Rules

```
New-PhishSimOverridePolicy -Name 'KnowBe4PhishOverridePolicy'  
$GUID = (Get-PhishSimOverridepolicy).guid  
New-PhishSimOverrideRule -Name 'KnowBe4PhishOverrideRule' -Policy $GUID  
-SenderDomainIs BigBox.Com,MediumBox.Com -SenderIpRanges 147.160.167.0/26,  
23.21.109.197,23.21.109.212
```

### Remove existing Phish Sim Override Policies and Rules

```
Remove-PhishSimOverridePolicy  
Remove-PhishSimOverrideRule
```

### Modify existing Phish Sim Override Policies and Rules

```
Set-PhishSimOverridePolicy -Identity $Guid -Comment 'New Phish Sim Override Policy'  
$RuleGuid = (Get-PhishSimOverrideRule).Guid  
Set-PhishSimOverrideRule -Identity $RuleGuid -RemoveSenderDomainIs  
'MediumBox.Com'  
Set-PhishSimOverrideRule -Identity $RuleGuid -RemoveSenderIpRanges '23.21.109.212'
```

**\*\* Note \*\*** Only one Phish Sim Over Ride Rule and Policy pair can be created in one tenant. Policy and Rule names will be overridden by default.

## Sec Ops Override

### List existing Phish Sim Policies or Rules

```
Get-SecOpsOverridePolicy  
Get-SecOpsOverrideRule
```

### Create New Phish Sim Override Policies and Rules

```
New-SecOpsOverridePolicy -Name SecTeamSecOpsOverridePolicy -SentTo  
SecurityTeam@powershellgeek.com  
New-SecOpsOverrideRule -Name SecOpsOverrideRule -Policy SecOpsOverridePolicy
```

### Remove existing Phish Sim Override Policies and Rules

```
$SecOpsPolicyGUID = (Get-SecOpsOverrideRule).GUID  
Remove-SecOpsOverridePolicy -Identity $SecOpsPolicyGUID  
$SecOpsRuleGUID = (Get-SecOpsOverrideRule).GUID  
Remove-SecOpsOverrideRule -Identity $SecOpsRuleGUID
```

### Modify existing Phish Sim Override Policies and Rules

```
Set-SecOpsOverridePolicy -Comment 'New Sec Ops Override Policy'  
Set-SecOpsOverrideRule -Comment 'New Sec Ops Override Rule'
```

**\*\* Note \*\*** Only one Phish Sim Over Ride Rule and Policy pair can be created in one tenant. Policy and Rule names will be overridden by default.

## Privacy Management

### List Privacy Management Case Admins

Get-PrivacyManagementCaseAdmin

### List Privacy Management Compliance Case Members

Get-PrivacyManagementComplianceCaseMember -Case 'SharePoint Privacy Case 01'

### List current Privacy Management Policies (matching scenario)

Get-PrivacyManagementPolicy -PrivacyManagementScenario OpenAccess

### List current Privacy Management Rules

Get-PrivacyManagementRule

### Create new Privacy Management Policy

New-PrivacyManagementPolicy -Name TestPMPolicy01 -PrivacyManagementScenario

HighRiskDataTransfer -ExchangeLocation All -Comment 'Test PM Policy 01' -Mode

TestWithNotifications

### Create new Privacy Management Rule

New-PrivacyManagementRule -Name 'Corp-DataTransfer-PM-Rule-02' -Policy

TestPMPolicy02 -SharepointSites All -PrivacyAccessLevel Public,Internal,External

### Remove an existing Privacy Case Admin

Remove-PrivacyManagementCaseAdmin damian@practicalpowershell.com

### Remove an existing Privacy Compliance Case Member

Remove-PrivacyManagementComplianceCaseMember -Case 'SharePoint Privacy Case 01'

-Member damian@powershellgeek.com

### Remove an Privacy Management Rule

Remove-PrivacyManagementPolicy TestPMPolicy01

### Remove an existing Privacy Management Rule

Remove-PrivacyManagementRule Corp-DataTransfer-PM-Rule-02

### Change settings on an existing Privacy Management Policy

Set-PrivacyManagementPolicy TestPMPolicy01 -Comment 'Created 2022-10-01 - Damian'

### Change settings on an existing Privacy Management Rule

Set-PrivacyManagementRule TestPMPolicy01 -Enabled \$False

### List existing Privacy Management Compliance Tags

Get-PrivacyManagementComplianceTag

### Create a new Privacy Management Compliance Tags

New-PrivacyManagementComplianceTag -Name CorpTag -RetentionDuration 365 -

RetentionAction Keep

### Remove an existing Privacy Management Compliance Tags

Remove-PrivacyManagementComplianceTag CorpTag

### Change settings on an existing Privacy Management Compliance Tag

Set-PrivacyManagementComplianceTag CorpTag -Comment 'PM Tag 2022'

## Connect to SCC with Certificates

<https://www.michev.info/Blog/Post/3796/connect-to-the-security-and-compliance-center-powershell-via-certificate-based-authentication>

## Adaptive Scope

### List existing Adaptive Scopes

Get-AdaptiveScope

### Create a new Adaptive Scope

New-AdaptiveScope -Name 'Chicago Executives' -LocationType User -FilterConditions @{"Conditions" = @(@{"Value" =

"Executive"; "Operator" = "StartsWith"; "Name" = "Title"},@{"Value" = "Chicago"; "Operator" = "Equals"; "Name" =

"Office"});"Conjunction" = "And"}

### Remove an existing Adaptive Scope

Remove-AdaptiveScope 'Chicago Executives'

### Change settings on an existing Adaptive Scope

Set-AdaptiveScope "US Users" -Comment 'For US users only'

## Miscellaneous

**View status of the Regulatory Compliance UI in Purview portal (preservation lock)**

Get-RegulatoryComplianceUI

**Use this to display regulatory record mark for retention label settings**

Set-RegulatoryComplianceUI

**Provide activity report of email encryption[10 Days]**

Get-MailDetailEncryptionReport

**Provide activity report of email encryption[92 Days]**

Get-MailTrafficEncryptionReport

## New Cmdlets – No Information

Get-AadProtectionLevel

Thank you to all that downloaded this cheatsheet!  
Feedback? [feedback@practicalpowershell.com](mailto:feedback@practicalpowershell.com)