Quest

Quest® Change Auditor 7.5

PowerShell Command Guide

If you have any questions regarding your potential use of this material, contact:
Quest Software Inc.
Attn: LEGAL Dept.
20 Enterprise, Suite 100
Aliso Viejo, CA 92656

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

Legend

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

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

Change Auditor PowerShell Command Guide
Updated - January 2025
Software Version - 7.5

Contents

PowerShell Commands 6
Adding the PowerShell module 7
Viewing available commands and help 7
Installing Change Auditor coordinators and web clients 8
Install-CACoordinator 8
Install-CAWebClient 9
Install-CALicense 9
Setting the master time zone 10
Set-CAScheduleMasterTimeZone 10
Get-CAScheduleMasterTimeZone 11
Finding Change Auditor installations and coordinators 11
Find-CAInstallations 11
Find-CACoordinators 12
Find-CASuitableCoordinator 12
Connecting to and disconnecting from Change Auditor installations and coordinators 13
Connect-CAClient 13
Disconnect-CAClient 15
Importing and exporting configuration settings 15
Import-CAConfigurations 15
Export-CAConfigurations 16
Managing client authentication options 16
Get-CAAuthenticationOptions 17
Set-CAAuthenticationOptions 17
Gathering Change Auditor system information 18
Get-CACoordinator 18
Get-CACoordinators 19
Get-CAInstallation 19
Get-CAAgents 19
Deploying Change Auditor agents 20
Install-CAAgent 20
Ping-CAAgent 21
Uninstall-CAAgent 21
Update-CAAgent 21
Update-CAAgentConfigurations 21
Set-CAAgentConfiguration 22
Get-CAAgentSubsystems 22
Enable-CAAgentTemplate 22
Disable-CAAgentTemplate 23
Remove-CAAgentTemplate 23
New-CAConfiguration 23
Get-CAConfigurations 24
Set-CAConfiguration 24
Remove-CAConfiguration 25

Managing auditing templates . . . 25
Add-CATemplateToConfiguration . . . 25
Get-CAConfigurationTemplates . . . 25
Get-CATemplatesInConfiguration . . . 26
Remove-CATemplatesFromConfiguration . . . 26
Working with searches . . . 27
Invoke-CASearch . . . 28
Get-CASearches . . . 28
Get-CASearchDefinition . . . 28
Set-CASearchProperties . . . 29
Copy-CASearch . . . 29
Add-CASearch . . . 30
Move-CASearch . . . 31
Remove-CASearch . . . 31
Add-CASearchFolder . . . 32
Remove-CASearchFolder . . . 32
Managing Active Directory Database auditing . . . 33
New-CAADDatabaseTemplate . . . 33
Get-CAADDatabaseTemplates . . . 33
Remove-CAADDatabaseTemplate . . . 34
Set-CAADDatabaseTemplate . . . 34
Working with Active Directory Database protection templates . . . 34
New-CAADDProtectionTemplate . . . 35
Set-CAADDProtectionTemplate . . . 35
Get-CAADDProtectionTemplates . . . 35
Remove-CAADDProtectionTemplate . . . 36
Managing Windows File System auditing . . . 37
New-CAWindowsFSAuditObject . . . 37
New-CAWindowsFSAuditTemplate . . . 39
Remove-CAWindowsFSAuditTemplate . . . 39
Set-CAWindowsFSAuditTemplate . . . 40
Get-CAWindowsFSAuditTemplates . . . 41
Get-CAWindowsFSEventClassInfo . . . 41
Managing SQL Extended Events Auditing (Preview) . . . 42
Get-CASQLExtendedEventsInfo . . . 42
New-CASQLExtendedEventsFilter . . . 43
New-CASQLExtendedEventsObject . . . 43
New-CASQLExtendedEventsTemplate . . . 44
Get-CASQLExtendedEventsTemplates . . . 45
Remove-CASQLExtendedEventsTemplate . . . 46
Managing Microsoft Entra ID auditing . . . 47
New-CAAzureADTemplate . . . 48
Set-CAAzureADTemplate . . . 52
Get-CAAzureADTemplates . . . 54
Managing Office 365 auditing . . . 54
New-CAO365Template . . . 55
Set-CAO365Template . . . 59

Get-CAO365Templates 62
Remove-CAO365Template 63
Get-CAO365ExchangeMailboxes 63
Add-CAO365ExchangeTemplateMailboxes 64
Remove-CAO365ExchangeTemplateMailboxes 64
Get-CAO365ExchangeTemplateMailboxes 65
Configuring a Quest On Demand Audit integration 65
New-CAODAConfiguration 66
Get-CAODAConfiguration 66
Set-CAODAConfiguration 67
Working with Active Directory protection templates 67
New-CAADProtectionTemplate 68
New-CAProtectedObject 68
Remove-CAProtectedObject 69
New-CAForestCredential 70
New-CAScheduledTimeRange 70
Get-CAADProtectionTemplates 70
Remove-CAADProtectionTemplate 72
Set-CAADProtectionTemplate 73
Working with GPO protection templates 74
New-CAGPOProtectionTemplate 74
Get-CAGPOProtectionTemplates 75
Set-CAGPOProtectionTemplate 76
Remove-CAGPOProtectionTemplate 77

About us 78
Our brand, our vision. Together. 78
Contacting Quest 78
Technical support resources 78

1

PowerShell Commands

Adding the PowerShell module

Viewing available commands and help

Installing Change Auditor coordinators and web clients

Setting the master time zone

Finding Change Auditor installations and coordinators

Connecting to and disconnecting from Change Auditor installations and coordinators

Managing client authentication options

Gathering Change Auditor system information

Deploying Change Auditor agents

Managing auditing templates

Working with searches

Managing Active Directory Database auditing

Working with Active Directory Database protection templates

Managing Windows File System auditing

Managing SQL Extended Events Auditing (Preview)

Managing Microsoft Entra ID auditing

Managing Microsoft Entra ID auditing

Managing Office 365 auditing

Configuring a Quest On Demand Audit integration

Configuring a Quest On Demand Audit integration

Working with Active Directory protection templates

Adding the PowerShell module

Change Auditor comes with a PowerShell module for you to use to manage your environment. It is installed when you install the Windows client or a coordinator.

NOTE: Windows PowerShell version 3.0 or higher is required.

To import the Change Auditor PowerShell module:

1. Open a Windows PowerShell window and type the following at the Windows PowerShell command prompt:

   ```
   Import-Module <path>
   ```

   Where "<path>" is the file path for the ChangeAuditor.PowerShell.dll assembly found in the Change Auditor Windows client or Change Auditor coordinator folder.

2. To ensure that the module was added, type the following at the Windows PowerShell command prompt:

   ```
   Get-Module -All
   ```

   The registered PowerShell modules are listed.

Viewing available commands and help

- To view all available Change Auditor commands, enter:

  ```
  Get-Command -Module ChangeAuditor.PowerShell
  ```

- To view help on each command including the syntax, enter:

  ```
  Get-Help cmdletName
  ```

- To view an interactive command browser that shows you the layout of commands and the help for the commands, enter:

  ```
  Show-Command cmdletName
  ```

NOTE: Sample scripts are available in the Change Auditor client folder. By default they are located here: C:\Program Files\Quest\ChangeAuditor\Client\PowerShell Sample Scripts

Installing Change Auditor coordinators and web clients

The following commands allow you to install Change Auditor components.

- Install-CACoordinator
- Install-CAWebClient
- Install-CALicense

Install-CACoordinator

Use this command to install locally a Change Auditor Coordinator.

Table 1. Available parameters

Parameter	Description
-MsiPath	The location to find the coordinator MSI file. The coordinator is installed using this installer.
-SQLAuthDatabaseCredential	Credentials to use for the coordinator to access the SQL Server. Specify when the coordinator should use SQL Authentication mode. **NOTE:** For Azure SQL Managed Instance (PaaS), SQL Authentication Database Credential or Microsoft Entra ID Credential must be used. The Encrypt connection option will be enabled by default.
-AzureADAuthDatabaseCredential	Credentials to use for the coordinator to access the SQL Server. Specify when the coordinator should use Microsoft Entra authentication mode. **NOTE:** For Azure SQL Managed Instance (PaaS), SQL Authentication Database Credential or Microsoft Entra ID Credential must be used. The Encrypt connection option will be enabled by default.
-DatabaseCredential	Credentials to use for the coordinator to access the SQL Server. Specify when the coordinator should use Windows Authentication mode. These credentials must be a valid set of Windows credentials..
-DatabaseServer	The SQL Server to host the database.
-LogPath	The local path on the computer where the installation log is written.
-AgentPort (Optional)	The static port for Change Auditor 6.x agents to communicate with the coordinator.
-ClientPort (Optional)	The static port for the Change Auditor client to communicate with the coordinator.
-DatabaseName (Optional)	The name assign to the Change Auditor database.
-InstallationName (Optional)	Name that uniquely identifies the current Change Auditor installation within your Active Directory environment. If this is an additional coordinator in an existing installation (sharing the same database), ensure that you use the name of the existing installation.
-LegacyAgentPort (Optional)	The static port for legacy (5.x) Change Auditor agents to communicate with the coordinator.
-SDKPort (Optional)	The static port used by external applications to access the coordinator

Example: Perform a local installation of a Change Auditor coordinator

```
Install-CACoordinator  -MsiPath "C:\Users\Administrator\Desktop\Quest Change Auditor
Coordinator 6 (x64).msi"  -SQLAuthDatabaseCredential $dbcredential -DatabaseServer
"MyDatabase" -LogPath "C:\Users\Administrator\Desktop\Coordinator.log"
```

After running this command, the installed coordinator will have the installation name “DEFAULT” and look for or create a database named ChangeAuditor.

Install-CAWebClient

Use this command to install locally the web client.

Table 2. Available parameters

Parameter	Description
-LogPath	The local path on the computer where the installation log is written.
-MsiPath	The location to find the web client MSI file. The web client is installed using this installer.
-CoordinatorConnection (Optional)	A previously created connection from Connect-CAClient.
-SiteName (Optional)	The web site name for the Change Auditor web client.
-SitePort (Optional)	A unique port for the web site to avoid conflicts with other IIS applications (for example, SharePoint® uses the default port 80; therefore, the IIS web site for the Change Auditor web client must use a different port). If a conflicting port is specified, attempting to launch the web client displays either an ‘HTTP 404 Not Found’ or ‘Page cannot be displayed’ error.

Example: Install a web client

```
Install-CAWebClient  -MsiPath "C:\Users\Administrator\Desktop\Quest Change Auditor
Web Client 6 (x64).msi" -CoordinatorConnection $connection -LogPath
"C:\Users\Administrator\Desktop\WebClientInstallationLog.log"
```

Install-CALicense

Use this command to install licenses to the coordinators in a Change Auditor installation.

Table 3. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-LicensePath	The license file directory on the client computer.
-Coordinator (Optional)	The single coordinator where you want to install the license (rather than all coordinators).
-Certificate (Optional)	The thumbprint of a client authentication certificate to use instead of automatically searching the current user's personal certificate store.

Example: Install a Change Auditor license in Installation domain, when using certificate authentication

```
$creds = Get-Credential

$connection = Connect-CAClient -credential $creds
```

```
$coordinatorToLicense = Get-CACoordinators -Connection $connection
Install-CALicense $connection –LicensePath 'c:\ChangeAuditor.dlv'
```

Example: Install license to single coordinator in the Installation which is installed on the local system, when using certificate authentication

```
$creds = Get-Credential
$connection = Connect-CAClient -SelectLocalCoordinator -credential $creds
$coordinatorToLicense = Get-CACoordinator -Connection $connection
Install-CALicense -Connection $connection -Coordinator $coordinatorToLicense –
LicensePath 'c:\ChangeAuditor.dlv'
```

Example: Install a Change Auditor for Active Directory license

```
Install-CALicense $connection –LicensePath C:\7_0_AD_license_PER.dlv
```

Example: Install license to single coordinator in the Installation which is installed on the local system

```
$connection =  Connect-CAClient -SelectLocalCoordinator
$coordinatorToLicense = Get-CACoordinator -Connection $connection
Install-CALicense -Connection $connection -Coordinator $coordinatorToLicense -
LicensePath 'c:\7_0_AD_license_PER.dlv'
```

NOTE: Optional Coordinator parameter object must be obtained using Get-CACoordinator command.

Setting the master time zone

Starting with version 6.9, Change Auditor calculates the Next Run of the reports, and archive and purge jobs based on the master time zone. For new deployments, the master time zone is set to the time zone of the server where the first coordinator is being installed. During an upgrade, the master time zone is set to UTC. You can manually change the master time zone, using the set-CAScheduleMasterTimeZone and get-CAScheduleMasterTimeZone commands. We recommend that you set the master time zone to the time zone where most the users are located.

NOTE: Because Daylight Saving Time changes on different dates worldwide, Change Auditor's schedules follow the time change of that specific time zone.

- Set-CAScheduleMasterTimeZone
- Get-CAScheduleMasterTimeZone

Set-CAScheduleMasterTimeZone

Use this command to specify which time zone the coordinators should use to calculate Next Run of the reports and archive and purge jobs.

Table 4. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TimeZoneID	The identifier of a system time zone.
-TimeZoneInfo	A TimeZoneInfo object.

NOTE: The TimeZoneID and TimeZoneInfo parameters must be a system-recognized time zone obtained through a call to the PowerShell command "[System.TimeZoneInfo]::GetSystemTimezones()".

Example: Set the schedule master time zone with a time zone info object

```
$atlanticTime = [System.TimeZoneInfo]::GetSystemTimeZones() |? {$_.Id -eq "Atlantic Standard Time"}

Set-CAScheduleMasterTimeZone -Connection $connection -TimeZoneInfo $atlanticTime
```

Example: Set the schedule master time zone with a time zone identifier

```
Set-CAScheduleMasterTimeZone -Connection $connection -TimeZoneId "Eastern Standard Time"
```

Get-CAScheduleMasterTimeZone

Use this command to retrieve what time zone the coordinators should use to calculate Next Run of the reports and archive and purge jobs.

Table 5. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Retrieve the schedules master time zone

```
Get-CAScheduleMasterTimeZone -Connection $connection
```

Finding Change Auditor installations and coordinators

The following commands allow you to find the Change Auditor installations and coordinators available in your Active Directory environment. Once connected, you can run additional commands to manage the deployment.

NOTE: The installations and coordinators that a search returns depends on your credentials and domain trusts.

- Find-CAInstallations
- Find-CACoordinators
- Find-CASuitableCoordinator

Find-CAInstallations

Use this command to search Active Directory for all available Change Auditor installations. The default is the current computer's forest, however, you can optionally specify a domain to search cross-forest for deployments.

NOTE: This command runs in the context of the current user running PowerShell. The user must have permission to search Active Directory in the specified domain.

Example: Find all Change Auditor installations in DomainName.com

```
Find-CAInstallations –DomainName 'DomainName.com'
```

Find-CACoordinators

Use this command to search Active Directory for all available coordinators. The default is the current computers forest, however, you can optionally specify a domain to search cross-forest for deployments. This search returns all the information required to connect to the coordinator including ports.

NOTE: This command runs in the context of the current user running PowerShell. The user must have permission to search Active Directory in the specified domain.

Example: Find all available coordinators in DomainName.com

```
Find-CACoordinators -DomainName 'DomainName.com'
```

Find-CASuitableCoordinator

Use this command to search Active Directory for a coordinator to which a connection can be made. The default is the current computers forest; however, you can optionally specify a domain to search cross-forest for deployments.

If more than one Change Auditor installation is discovered, the call fails and the –InstallationName parameter is required.

Table 6. Available parameters

Parameter	Description
-Certificate (Optional)	When certificate authentication between the client and coordinator in environments is in place, this parameter specifies the thumbprint string copied from a certificate found in the certmgr.msc Certificate Manager for the current user in the Personal\Certificates folder. This certificate must be trusted, not expired, have a private key and at least the “Client Authentication” purpose. (“Proves your identity to a remote computer”).

Example: Find a coordinator in ‘DEFAULT’ installation that you have the credentials to connect to

```
Find-CASuitableCoordinator –InstallationName ‘DEFAULT’
```

Example: Find a coordinator in Domain that you have the credentials to connect to, when using certificate authentication

```
$creds = Get-Credential
$connection = Connect-CAClient -Credential $creds
Find-CASuitableCoordinator -Credential $creds -DomainName ‘DomainName.com'
```

Connecting to and disconnecting from Change Auditor installations and coordinators

- Connect-CAClient
- Disconnect-CAClient

Connect-CAClient

Most Change Auditor commands require a connection to a coordinator. This connection can be assigned to a variable and used for any command that requires it. This command searches for a suitable coordinator in a Change Auditor installation and creates a connection. Suitable coordinators are those to which you have access to and can be located by searching through Active Directory service connection points.

You can also connect to Change Auditor installations in untrusted domains or to a specific coordinator by specifying the -ComputerName and -Port parameters.

You can make multiple connections to different coordinators or deployments in the same script as long as the version of Change Auditor is the same.

NOTE: Connections are closed when the PowerShell session is ended or disconnected.

Table 7. Available parameters

Parameter	Description
-Credential (Optional)	Windows credentials specifying the user to connect to the Change Auditor installation. All operations using this connection will be authorized as this user. When not specified, the current client running PowerShell is used. **NOTE:** Credentials are required when certificate authentication is being used.
-CoordinatorConnectionPoint	Specify to use a specific coordinator found from a previous call to Find-CACoordinators.
-SelectLocalCoordinator	Create a connection to the local coordinator.
-InstallationName (Optional)	The installation name to connect to. If an installation cannot be found with this name, no connection is made. If more than one Change Auditor installation exists in the current forest, this parameter is mandatory. Omitting it results in a connection failure due to ambiguity.
-DomainName (Optional)	The name of the domain where the Change Auditor installation exists.
-ComputerName	The computer to connect to.
-Port	The port to connect to.
-WaitForServiceReady (Optional)	The number of seconds to wait for the connected coordinator service to be ready. **NOTE:** If not specified, when the Change Auditor coordinator is not ready for connections due to an in-progress install or upgrade, an error is returned. The maximum is 144,000 seconds, or 10 hours.
-UseCertificateAuth (Optional)	This parameter specifies that the coordinator is expected to be configured for certificate authentication. This parameter does not require an input value. When specified, this parameter is $true.
-DisableCertificateCrlCheck (Optional)	When certificate authentication between the client and coordinator in environments is in place, this parameter specifies that Certificate Revocation List checking is disabled. This parameter does not require an input value. When specified, this parameter is $true.
-Certificate (Optional)	When certificate authentication between the client and coordinator in environments is in place, this parameter specifies the thumbprint string copied from a certificate found in the certmgr.msc Certificate Manager for the current user in the Personal\Certificates folder. This certificate must be trusted, not expired, have a private key and at least the “Client Authentication” purpose. (“Proves your identity to a remote computer”).

Table 8. Supported parameter sets that enable a connection

Example	Enter the following command:
Recommended: Connect to the installation “XYZ” in the local forest. **NOTE:** This allows for fault tolerance if you have numerous coordinators by selecting the best option in the domain.	`Connect-CAClient –InstallationName ‘XYZ’ -DomainName 'DomainName.com'`
Connect to the first suitable coordinator found in any installation in any trusted domain.	`$connection = Connect-CAClient`

Table 8. Supported parameter sets that enable a connection

Example	Enter the following command:
Connect to a specific coordinator by computer name and port.	`Connect-CAClient -ComputerName 'ca-cord.DomainName.com' -Port 52289`
Connect to the first suitable coordinator in the domain "DomainName.com".	`Connect-CAClient -DomainName 'DomainName.com'`
Connect to the first suitable coordinator in the domain "DomainName.com" with an installation name "DEFAULT".	`Connect-CAClient -DomainName 'DomainName.com' -InstallationName 'DEFAULT'`
Connect to a coordinator found from Find-CACoordinators.	`$coordinators = Find-CACoordinators -DomainName 'DomainName.com'` `$connection = Connect-CAClient -CoordinatorConnectionPoint $coordinators[0]`
Connect to a specific coordinator by computer name and port, when using certificate authentication.	`$creds = Get-Credential` `Connect-CAClient -Credential $creds -ComputerName 'ca-coordinator.DomainName.com' -Port 52289 -UseCertificateAuth`

Disconnect-CAClient

Use this command to disconnect from Change Auditor. (This is the equivalent of closing the Change Auditor client.)

Example: Connect to a Change Auditor deployment, and then close the connection

```
$connection = Connect-CAClient -InstallationName 'DEFAULT'

# perform some actions

Disconnect-CAClient $connection
```

Importing and exporting configuration settings

- Import-CAConfigurations
- Export-CAConfigurations

Import-CAConfigurations

Use this command to import Change Auditor configuration settings.

NOTE: All settings contained in the specified import file are imported by default when using this command except for Application User Interface and Coordinator configurations which can be included by using the provided optional parameters.

Table 9. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ConfigurationSettings	An XML document (Xml.XmlDocument object) that contains the configuration settings to import.
-IncludeCoordinatorConfig (Optional)	Imports the coordinator configuration if present in the exported file. **NOTE:** • This will overwrite existing settings, including SMTP Mail Server and Alert settings. • Scheduled task settings will not be applied when selected coordinators are missing.
-IncludeAppUserInterfaceConfig (Optional)	Imports the Application User Interface configuration. **NOTE:** This will overwrite existing coordinator user role settings and assignments which may require a restart of the coordinator in order to regain access.
-Force (Optional)	Suppresses warning y/n prompts from the -IncludeCoordinatorConfig and -IncludeAppUserInterfaceConfig parameters.

Example: Importing configuration settings

```
$connection = Connect-CAClient -Credential $creds
[xml] $xmld = Get-Content 'C:\DataFolder\CAConfig.xml'
Import-CAConfigurations -Connection $connection -ConfigurationSettings $xmld
```

Export-CAConfigurations

Use this command to export Change Auditor configuration settings.

NOTE: All settings that can be exported with the Windows client interface are included by default when exporting configuration settings using this command, including settings for Application User Interface and Coordinator configurations.

Table 10. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Exporting Change Auditor configuration settings

```
$connection = Connect-CAClient -Credential $creds
$xmlDoc = Export-CAConfigurations $connection
$xmlDoc.Save("C:\Configurations\CAconfig.xml")
```

Managing client authentication options

Change Auditor has two authentication method:

- Windows Forms Authentication (enabled by default)

  When users log in, they must enter a Windows user account and a password.

- Active Directory Client Certificate Authentication

  When users log in, they must specify a smart card or certificate. User account and password are not required.

These commands allow you to manage the authentication used in your Change Auditor deployment.

Get-CAAuthenticationOptions

Use this command to view the authentication profile Change Auditor coordinators use in a particular installation.

Returns: An object containing the options for authentication for the specified installation.

Table 11. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-InstallationName (Optional)	The installation name to connect to. If an installation cannot be found with this name, no connection is made. If more than one Change Auditor installation exists in the current forest, this parameter is mandatory. Omitting it results in a connection failure due to ambiguity.
-DomainName (Optional)	The name of the domain where the Change Auditor installation exists.

Example

```
Get-CAAuthenticationOptions –InstallationName ‘DEFAULT’ –DomainName ‘DomainName.com’
Get-CAAuthenticationOptions -Connection $connection
```

Set-CAAuthenticationOptions

Use this command to alter the authentication profile the Change Auditor coordinators use in a particular installation.

Returns: An object containing the options for authentication for the specified installation.

Table 12. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-AlwaysChallengeForCredential (Optional)	When specified, instructs the coordinator to disallow any connection that is not accompanied by credentials. For PowerShell clients, this means that the Connect-CAClient command will not connect without the use of the -Credential parameter.
-AllowActiveDirectoryCertificateAuthentication (Optional)	When specified, instructs the coordinator to allow certificate authentication via a web client. This switch has no meaning for the Win32 client.
-AllowWindowsFormsAuthentication (Optional)	When specified, instructs the coordinator to accept default username/password style of credentials.
-AuthenticationOptions (Optional)	This parameter allows the caller to pass directly the result of the Get-CAAuthenticationOptions without having to break down the options into their constituent flag values.

Example

```
Set-CAAuthenticationOptions -Connection $connection -AlwaysChallengeForCredential
-AllowActiveDirectoryCertificateAuthentication -AllowWindowsFormsAuthentication
```

```
Set-CAAuthenticationOptions -Connection $connection -AuthenticationOptions
$AuthenticationOptions
```

Gathering Change Auditor system information

You can gather Change Auditor system information to help you to manage your installation components.

- Get-CACoordinator
- Get-CACoordinators
- Get-CAInstallation
- Get-CAAgents

Get-CACoordinator

Use this command to retrieve coordinator-specific (as opposed to installation-wide) status information from the connected coordinator such as coordinator name, status, deployment name, version, connected agents, connected legacy agents, connected clients, client port, total events, and buffered events which may be different on each coordinator.

Example: Gather coordinator information for a specified connection

```
Get-CACoordinator $connection
```

Example: Gather coordinator information for a specified connection, when using certificate authentication

```
$creds = Get-Credential
```

```
$connection = Connect-CAClient -Credential $creds
Get-CACoordinator -Connection $connection
```

Get-CACoordinators

Use this command to gather information about all the coordinators in a Change Auditor installation.

Example: Gather coordinator information for all coordinators for a specified connection

```
Get-CACoordinators -Connection $connection
```

Example: Gather coordinator information for all coordinators for a specified connection, when using certificate authentication

```
$creds = Get-Credential
$connection = Connect-CAClient -Credential $creds
Get-CACoordinators -Connection $connection
```

Get-CAInstallation

Use this command to retrieve installation-specific (as opposed to coordinator-specific) status information including the name of the installation, database server, and database and the database size.

Example: Gather installation information for a specified connection

```
Get-CAInstallation -Connection $connection
```

Get-CAAgents

Use this command to view information on all available (and optionally uninstalled) agents.

NOTE: This returns information for workstation, server, and domain controller agents.

Table 13. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-IncludeUninstalled (Optional)	Adds uninstalled agents to the list of agents returned from this command.

Example: Viewing all available and unistalled agents within a specific installation

```
Get-CAAgents -Connection $connection -IncludeUninstalled
```

Deploying Change Auditor agents

The following commands are available to manage your agent deployments.

NOTE: You must be a member of the Administrators role to use these commands.

NOTE: Any changes affecting configuration are audited with internal events.

- Install-CAAgent
- Ping-CAAgent
- Uninstall-CAAgent
- Update-CAAgent
- Update-CAAgentConfigurations
- Set-CAAgentConfiguration
- Get-CAAgentSubsystems
- Enable-CAAgentTemplate
- Disable-CAAgentTemplate
- Remove-CAAgentTemplate
- New-CAConfiguration
- Get-CAConfigurations
- Set-CAConfiguration
- Remove-CAConfiguration

Install-CAAgent

Use this command to install an agent.

Table 14. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-MachineName	The fully qualified name of a target computer.
-Credential	Credentials used to access the target computer.
-OperationTime (Optional)	Specifies when to perform this operation. **NOTE:** If this is not specified, it defaults to the current time.

Example: Install an agent

```
Install-CAAgent -Connection $connection -MachineName "ComputerName.DomainName.com" -
Credential $credential -OperationTime “01/01/2020 12:00:00”
```

Ping-CAAgent

Use this command to ensure that the coordinator and agent can communicate using WCF framework.

Table 15. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-AgentInfo	The PSCAAgentInfo retrieved from the Get-CAAgents command.

Example: Test the communication between an agent and coordinator

```
Ping-CAAgent -Connection $connection -AgentInfo $agentinfo
```

Uninstall-CAAgent

Use this command to uninstall an agent.

Table 16. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-MachineName	The fully qualified name of the target computer.
-Credential	Credentials used to access the target computer.
-OperationTime (Optional)	Specifies when to perform this operation. **NOTE:** If this is not specified, it defaults to the current time.

Example: Uninstall an agent

```
Uninstall-CAAgent -Connection $connection -MachineName "ComputerName.DomainName.com"
-Credential $credential -OperationTime “01/01/2020 12:00:00”
```

Update-CAAgent

Use this command to upgrade an agent.

Table 17. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Agent	Agents obtained from a previous call to Get-CAAgents.
-Credential	Credentials used to access the target computer.
-OperationTime (Optional)	Specifies when to perform this operation. **NOTE:** If this is not specified, it defaults to the current time.

Example: Upgrade an agent

```
Update-CAAgent -Connection $connection -Agent $agent -Credential $credential
```

Update-CAAgentConfigurations

Use this command to update the agent configuration to ensure that the agent is using the most up-to-date configuration.

Table 18. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Agents	Agents obtained from a previous call to Get-CAAgents.

Example: Update an agent configuration

```
Update-CAAgentConfigurations -Connection $connection -Agents $agent
```

Set-CAAgentConfiguration

Use this command to assign an auditing configuration to an agent.

Table 19. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Agents	Agents obtained from a previous call to Get-CAAgents.
-Configuration	The configuration obtained by a previous call to Get-CAConfigurations.

Example: Update an agent configuration

```
Set-CAAgentConfiguration -Connection $connection -Agents $agent -Configuration
$configuration
```

Get-CAAgentSubsystems

Use this command to see the list of subsystems included in an agent‘s configuration.

Table 20. Available parameters

Parameter	Description
-AgentInfo	The PSCAAgentInfo retrieved from the Get-CAAgents command.

Example: See a list of all subsystems included in an agent‘s configuration

```
Get-CAAgentSubsystems -AgentInfo $agentinfo
```

Enable-CAAgentTemplate

Use this command to enable a template.

NOTE: Currently, this is only supported for Microsoft Entra ID and Microsoft 365.

Table 21. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to modify.
-Credential (Optional)	Credentials associated with the target agent and template. These vary depending on the type of template.

Example: Enable a template

```
Enable-CAAgentTemplate -Connection $connection -Template $template
```

Disable-CAAgentTemplate

Use this command to disable a template.

NOTE: Currently, this is only supported for Microsoft Entra ID and Microsoft 365.

Table 22. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to modify.
-Credential (Optional)	Credentials associated with the target agent and template. These vary depending on the type of template.

Example: Disable a template

Disable-CAAgentTemplate -Connection $connection -Template $template

Remove-CAAgentTemplate

Use this command to remove a template.

NOTE: Currently, this is only supported for Microsoft Entra ID and Microsoft 365.

Table 23. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to remove.
-Credential (Optional)	Credentials associated with the target agent and template. These vary depending on the type of template.

Example: Remove a template

```
Remove-CAAgentTemplate -Connection $connection -Template $template -credential
$credential
```

New-CAConfiguration

Use this command to create an agent configuration.

Table 24. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ConfigurationName	The name of the agent configuration to create.

Example: Create an agent configuration

```
New-CAConfiguration -Connection $connection -ConfigurationName $configurationName
```

Get-CAConfigurations

Use this command to get list of all agent configurations for a deployment.

Table 25. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: See a list of all agent configurations

```
Get-CAConfigurations -Connection $connection
```

Set-CAConfiguration

Use this command to change the agents port used for the coordinator to communicate with the agent and to configure a proxy server.

NOTE: If you change the agent port number, you must also create a firewall exception for the new port number on your agent computers.

NOTE: If your organization uses a proxy server to connect to the internet, you must configure the proxy parameters to audit Microsoft Entra ID and Microsoft 365 targets. If your proxy server requires authentication, you must also set the credentials using the -ProxyCredential parameter.

Table 26. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Configuration	The configuration on which to set the port on.
-Port	The port the agent starts its service on for coordinator and agent communication.
-ProxyServer	The fully qualified domain name, down-level name, or IPv4 address of the proxy server. **NOTE:** To clear the proxy configuration and set the proxy settings back to the default values, specify an empty value for this parameter.
-ProxyPort	The port on which to communicate with the proxy server. (Default is 8080).
-ProxyCredential	The credentials used to authenticate with the proxy server.
-ClearProxyCredential	Specify this parameter to clear the credentials for the proxy server authentication.

Example: Update the port used to communicate with the agent

```
Set-CAConfiguration -Connection $connection -Configuration $configurationObject -
Port $port
```

Example: Update the configuration to allow for cloud-based auditing

```
Set-CAConfiguration -Connection $connection -Configuration $config -ProxyServer
"ServerName" -ProxyPort 8080
```

Remove-CAConfiguration

Use this command to remove an existing agent configuration.

Table 27. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Configuration	The name of the configuration to remove. **NOTE:** You cannot delete the default configuration template.

Example: Remove an agent

```
Remove-CAConfiguration -Connection $connection -Configuration $configuration
```

Managing auditing templates

- Add-CATemplateToConfiguration
- Get-CAConfigurationTemplates
- Get-CATemplatesInConfiguration
- Remove-CATemplatesFromConfiguration

Add-CATemplateToConfiguration

Use this command to assign an auditing template to a Change Auditor configuration.

Table 28. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Configuration	The configuration to which to add a template. Use Get-CAConfigurations to obtain the configuration object.
-Templates	The templates to apply to the configuration. Use Get-CAConfigurationTemplates to obtain the templates.

Example: Assign a template to a configuration

```
Add-CATemplateToConfiguration -Connection $connection -Configuration $configuration
-Templates $templates
```

Get-CAConfigurationTemplates

Use this command to get a list of all templates in the installation.

Table 29. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get a list of all templates in the installation

```
Get-CAConfigurationTemplates -Connection $connection
```

Get-CATemplatesInConfiguration

Use this command to get a list of the templates that are assigned to a configuration.

Table 30. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Configuration	Use Get-CAConfigurations to obtain the configuration object.

Example: Get a list of all templates assigned to a configuration

```
Get-CATemplatesInConfiguration -Connection $connection -Configuration $configuration
```

Remove-CATemplatesFromConfiguration

Use this command to remove templates from a configuration.

Table 31. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Configuration	The configuration from which to remove a template. Use Get-CAConfigurations to obtain the configuration object.
-Templates	The templates to remove from the configuration. Use Get-CAConfigurationTemplates to obtain the templates.

Example: Remove a template from a configuration

```
Remove-CATemplatesFromConfiguration -Connection $connection -Connection $connection
-Configuration $configuration
```

Working with searches

Searches (both built-in and private) allow you to view valuable information based on activity captured by Change Auditor.

When using the commands, consider the following:

- You cannot create multiple folders with the same name in the same directory.
- Microsoft folder naming standards are upheld and restrict folder names to a length between 1 and 4000 characters.
- You cannot create multiple searches with the same name in the same directory.
- The commands generate audit events when a folder that contains public searches is deleted.
- Administrators have full access to all the search commands.
- Operators access inlcudes the following:

  Full access for:

  - Invoke-CASearch
  - Get-CASearches
  - Get-CASearchDefinition

  Restricted access to private searches and folders for:

  - Set-CASearchProperties
  - Copy-CASearch
  - Add-CASearch
  - Move-CASearch
  - Remove-CASearch
  - Add-CASearchFolder
  - Remove-CASearchFolder

The following commands are available to manage searches:

- Invoke-CASearch
- Get-CASearches
- Get-CASearchDefinition
- Set-CASearchProperties
- Copy-CASearch
- Add-CASearch
- Move-CASearch
- Remove-CASearch
- Add-CASearchFolder
- Remove-CASearchFolder

Invoke-CASearch

Use this command to run a search.

Table 32. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Search	The search to run. Use Get-CASearches to find the PSCASearchInfo object required to identify the search.
-StartTime (Optional)	The start time for the events that will be retrieved. By default this is the start time defined in the search.
-EndTime (Optional)	The end time for the events that will be retrieved. By default this is the start time defined in the search.
-Limit (Optional)	The maximum number of records to retrieve and display. By default this is the limit defined in the search.

Example: Running a search and limit the display to 10 events

```
$connection = Connect-CAClient -InstallationName 'DEFAULT"

$search = Get-CASearches $connection | ? {$_.Name -eq "All Events"}

Invoke-CASearch -Connection $connection -Search $search -limit 10
```

Get-CASearches

Use this command to view information on all available searches and identify a search info object that is required for some other commands.

Table 33. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Viewing all available searches within a specific installation

```
Get-CASearches $connection
```

Example: Viewing a specific search

```
Get-CASearches $connection | ? {$_.Name -eq "All AD Queries in the last 30 days"}
```

Get-CASearchDefinition

Use this command to obtain the search definition from an existing search. The search definition is XML that can be modified and used to create a search.

Table 34. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Search	The search info object obtained from the Get-CASearches command.

Example: Getting the definition of a search with the name “All Events” and writing it to a file at the directory “C:\definitions\All Events.xml”

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

[xml]$xmlString = Get-CASearches $connection | ? {$_.Name –eq “All Events”} | Get-CASearchDefinition $connection

$xmlString.Save(“C:\definitions\All Events.xml”)
```

Set-CASearchProperties

Use this command to update the search name, default folder, set the limit of a public or private search, or the path and subsystem for an imported .csv file of a list of directory objects.

Table 35. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Search	The search info object obtained from the Get-CASearches command.
-Name	Specifies a new name for the search.
-DefaultFolderPath	Specifies a new default folder path for the search.
-Limit	Specifies a new limit for the search.
-PassThru (Optional)	A switch that specifies to return the updated search after the command runs.
-Subsytem	The subsystem to update. The ability to import a .csv file with a list of objects is available for Active Directory, Exchange, and Group Policy.
-Path	Path to the .csv file to import.

Example: Changing the display name of a search, set the default folder path and limit

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$search = Get-CASearches $connection | ?{$_.Name –eq “All Owner Mailbox Events”}

Set-CASearchProperties   $connection  -Search $search -Name "NewName"
-DefaultFolderPath "C:\PATH\MYSEARCH" -Limit 1000
```

Example: Import a .csv file of Active Directory objects

NOTE: For optimal performance, do not include more than 1000 objects in your import file.

```
$connection=Connect-CAClient -InstallationName 'Default'

$search = Get-CASearches $connection | ? {$_.Name -eq "All My Events"}

Set-CASearchProperties $connection -Search $search -Subsystem "Active Directory" -
Path "C:\MyCSVObjectList.csv"
```

Copy-CASearch

Use this command to copy a search in the installation.

Table 36. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Search	The search info object obtained from the Get-CASearches command.

Table 36. Available parameters

Parameter	Description
-IsPublic (Optional)	An optional switch that specifies if the search is public. The default is private.
-UserSid	An optional parameter that is used (when –IsPublic is not used) to specify the SID of the user that owns the directory where the copy of the search is placed.
-Path	A parameter that specifies a path where the copy is to be placed. The default is the root folder of the user/public folder specified with –UserSid /-IsPublic.
-Name (Optional)	An optional parameter that specifies a new name for the copy of the search.
-PassThru (Optional)	A switch that specifies to return the updated search after the command runs.

Example: Copying a search named “New Search for Employee” to a user’s private folder Searches\New and giving it a new name “All My Events”

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$search = Get-CASearches $connection | ? {$_.Name –eq “New Search for Employee”}

Copy-CASearch –Connection $connection –Search $search –UserSid S-1-5-21-3623811015-
3361044348-30300820-1013 –Path Private\Searches\New –Name “All My Events” -PassThru
```

Add-CASearch

Use this command to create a search in the installation.

Table 37. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-XmlSearchDefinition	An XML string or object that represents a search definition.
-IsPublic	A switch that specifies if the search is public. The default is private.
-UserSid	A parameter that is used (when –IsPublic is not used) to specify the SID of the user who owns the new search.
-Path	A parameter that specifies a path where the new search will be placed. The default is the root folder of the user/public folder specified with –UserSid /-IsPublic.
-Name	A parameter that specifies a new name for the search.
-PassThru (Optional)	A switch that specifies to return the new search after the command runs.

Example: Adding a public search to the installation

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$searchDefinition = Get-Content C:\Users\Admin\Documents\MySearchDefinition.xml

Add-CASearch –Connection $connection –XmlSearchDefinition $searchDefinition
–IsPublic –Path Shared\AllSearches\New –Name “All events in the past 23 hours”
-PassThru
```

Move-CASearch

Use this command to move a search from one folder path to another in the installation.

Table 38. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-IsPublic	A switch that specifies if the search is public. The default is private.
-UserSid	A parameter that is used (when –IsPublic is not used) to specify the SID of the user who owns the new search.
-Path	A parameter that specifies the path where the search will be placed. The default is the root folder of the user/public folder specified with –UserSid /-IsPublic.
-Search	The search info object obtained from the Get-CASearches command.
-PassThru (Optional)	A switch that specifies to return the updated search after the command runs.

Example: Moving the search named “All AD Queries in the last 30 days” to the private folder “Shared\SharePoint\” of the user with the SID “S-1-5-21-3623811015-3361044348-30300820-1013”

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$search = Get-CASearches $connection | ? {$_.Name –eq “All AD Queries in the last 30 days”}

Move-CASearch $connection –Search $search –UserSid S-1-5-21-3623811015-3361044348-30300820-1013 –Path “Shared\SharePoint”
```

Remove-CASearch

Use this command to remove a public or private search from the installation.

Table 39. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Search	The search info object obtained from the Get-CASearches command.
-Force (Optional)	A parameter that removes the prompt before a search is removed.

Example 1: removing any search with the name “All Exchange Admin Events” from the installation

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$search = Get-CASearches $connection | ? {$_.Name –eq “All Exchange Admin Events”}

Remove-CASearch $connection –Search $search
```

Example 2: Removing the search with the name “All Search Events”, owned by the user with the SID “S-1-5-21-3623811015-3361044348-30300820-1013”, which exists in that user’s folder “Security\Internal\Searches” from the installation

```
$connection = Connect-CAClient –InstallationName ‘DEFAULT’

$search = Get-CASearches $connection | ? {$_.OwnerSid –eq “S-1-5-21-3623811015-3361044348-30300820-1013”} | ? {$_.FolderPath –eq “Security\Internal\Searches”} | ? {$_.Name –eq “All Search Events”}
```

```
Remove-CASearch $connection –Search $search
```

Add-CASearchFolder

Use this command to create a search folder in the installation.

Table 40. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-IsPublic	A switch that specifies if the search is public. The default is private.
-UserSid	A parameter that is used (when –IsPublic is not used) to specify the SID of the user who owns the new folder.
-Path	A parameter that specifies the path to create. The default is the root folder of the user/public folder specified with –UserSid /-IsPublic.

Example: Adding the public folder Searches\New to the installation

```
$connection = Connect-CAClient –InstallationName 'DEFAULT'
Add-CASearchFolder –Connection $connection –IsPublic –Path Shared\Searches\New
```

Remove-CASearchFolder

Use this command to remove a public or private folder from the installation.

Table 41. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-IsPublic	A switch that specifies the folder being removed is public.
-UserSid	A parameter that is used if –IsPublic is not specified to speci-fy the SID of the user that owns the private folder being removed.
-Path	A parameter that specifies the path to the folder to remove. The default is the root folder of the user/public folder specified with –UserSid /-IsPublic.
-Force (Optional)	An optional parameter that removes the prompt before a search is removed.

Example: Removing the public folder in the installation Miscellaneous\OldSearches

```
$connection = Connect-CAClient –InstallationName 'DEFAULT'
Remove-CASearchFolder $connection –IsPublic –Path Shared\Miscellaneous\OldSearches
```

Managing Active Directory Database auditing

Change Auditor allows you to monitor the Active Directory database (NTDS.dit) file for possible unauthorized access attempts.

Extraction of this file could lead to parsing of usernames and passwords resulting in a security breach. The ability to audit changes to this file reduces the risk of the user account information from being accessed and tampered with by unwanted processes or users.

Managing Active Directory database auditing is available through the following PowerShell commands:

- New-CAADDatabaseTemplate
- Get-CAADDatabaseTemplates
- Remove-CAADDatabaseTemplate
- Set-CAADDatabaseTemplate

New-CAADDatabaseTemplate

Use this command to create an Active Directory Database auditing template.

Table 42. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TemplateName	The template name.
-Disabled (Optional)	Set to true or false to enable or disable the template.
-ExcludedProcesses (Optional)	The list of processes to exclude from auditing. The default is none.

Example: Create a new Active Directory Database auditing template

```
New-CAADDatabaseTemplate -Connection $connection -TemplateName $template
-ExcludeProcess $excludeProcess -Disabled false
```

Get-CAADDatabaseTemplates

Use this command to see all the Active Directory Database auditing templates available within your installation.

Table 43. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Id (Optional)	The template GUID.

Example: Get a list of all Active Directory Database templates

```
Get-CAADDatabaseTemplates -Connection $connection
```

Remove-CAADDatabaseTemplate

Use this command to delete an Active Directory Database auditing template.

Table 44. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The CAADDatabaseTemplate object to remove. Obtain the template objects using the Get-CAADDatabaseTemplates command and filter to select the object to remove.
-Force (Optional)	Removes template without prompting for a confirmation. The default is false.

Example: Remove a Active Directory Database auditing template

```
Remove-CAADDatabaseTemplate -Connection $connection -Template $removeTemplate
```

Set-CAADDatabaseTemplate

Use this command to modify an Active Directory Database auditing template.

Table 45. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TemplateName	The new name that you want to use for the template.
-Template	The PSCAProtectionTemplate object to update. Obtain the template objects using the Get-CAADDatabaseTemplates command and filter to select the object to update.
-Disabled (Optional)	Set to $true or $false to disable or enable the template respectively.
-ExcludedProcesses (Optional)	The list of processes to exclude from auditing. The default is none.

Example: Modify an Active Directory Database auditing template

```
Set-CAADDatabaseTemplate -Connection $connection -template $template -templatename
"Name" -ExcludeProcess $excludeProcess -Disabled $false
```

Working with Active Directory Database protection templates

Change Auditor allows you to protect the Active Directory database (NTDS.dit) file for possible unauthorized access attempts.

The following commands are available to manage Active Directory Database protection:

- New-CAADDProtectionTemplate
- Set-CAADDProtectionTemplate
- Get-CAADDProtectionTemplates
- Remove-CAADDProtectionTemplate

New-CAADDProtectionTemplate

Use this command to create an Active Directory Database protection template.

Table 46. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TemplateName	The template name.
-Disabled (Optional)	Set to true or false to enable or disable the template.
-ExcludedProcesses (Optional)	The list of processes to exclude from protectoin. The default is none.

Example: Create an Active Directory Database protection template

```
New-CAADDProtectionTemplate -Connection $connection -TemplateName TemplateSample
```

Set-CAADDProtectionTemplate

Use this command to modify an Active Directory Database protection template.

Table 47. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The CAADDProtectionTemplate object to edit. Obtain the template objects using the Get-CAADDProtectionTemplates command and filter to select the object to update.
-TemplateName (Optional)	The template name.
-Disabled (Optional)	Set to true or false to enable or disable the template.
-ExcludedProcesses (Optional)	The list of processes to exclude from protectoin. The default is none.

Example: Create an Active Directory Database protection template

```
set-caaddprotectiontemplate -connection $connection -template $template -
templatename "templatesample"
```

Get-CAADDProtectionTemplates

Use this command to see all the Active Directory Database protection templates that have been created.

Table 48. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ID (Optional)	GUID for a specific template.

Example: Get a list of all Active Directory Database Protection templates

```
Get-CAADDProtectionTemplates -Connection $connection
```

Remove-CAADDProtectionTemplate

Use this command to remove an Active Directory Database protection template.

Table 49. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The PSCAProtectionTemplate object to remove. Obtain the template objects using the Get-CAADDatabaseTemplates command and filter to select the object to remove.
-Force (Optional)	Removes the template without providing confirmation.

Example: Remove an Active Directory Database protection template

```
Remove-CAADProtectionTemplate -Connection $connection -Template $template
```

Managing Windows File System auditing

Change Auditor for Windows File Server tracks, audits, and alerts on file and folder changes in real time, translating events into simple terms and eliminating the time and complexity required by system provided auditing. The auditing scope can be set on an individual file or folder or an entire file system recursive or non-recursive. You can include or exclude certain files or folders from the audit scope to ensure a faster and more efficient audit process.

Managing Windows file system auditing is available through the following PowerShell commands:

- New-CAWindowsFSAuditObject
- New-CAWindowsFSAuditTemplate
- Remove-CAWindowsFSAuditTemplate
- Set-CAWindowsFSAuditTemplate
- Get-CAWindowsFSAuditTemplates
- Get-CAWindowsFSEventClassInfo

New-CAWindowsFSAuditObject

Use this command to define a folder or file paths to audit.

NOTE:

- For All Drives - IncludePath is '*', IncludePathType is Folder and IncludeScope is ScopeOneLevel or ScopeSubtree.
- When IncludePath is set to '[All Shares]', this is a SystemShare.

Table 50. Parameter description

Parameter	Description
-IncludePath	Specifies the folder or file to audit. **NOTE:** Built-in folder values include Common Program Files, Program Files, System Drive, Windows Directory, and All Shares.
-IncludePathType	Specifies the type of path to audit based on one of the following values: • SystemFile • SystemFolder • SystemShare **NOTE:** Only one type of path can be specified.
-IncludeScope	Specifies the scope to monitor for the Includepath based on one of the following values: • ScopeObject • ScopeOneLevel • ScopeSubtree
-AuditEvents	The events to audit. Use Get-CAWindowsFSEventClassInfo to get the list of event classes.
-IncludeMask (Optional)	Specifies what to include in the selected folder or file path to audit. Entering * will audit all files and folders in the selected folder. **NOTE:** Includemask is required for Systemfolder and systemshare types but not systemfile.
-ExcludeFilePaths (Optional)	Specifies the names and paths of any files to exclude from auditing. The default is set to None.
-ExcludeFolderPaths (Optional)	Specifies the names and paths of any subfolders to exclude from auditing. The default is set to None.
-Disabled (Optional)	Specifies whether auditing is enabled or disabled on the selected path or folder. The default is set to false.

Example: Monitoring a directory for all file types and all subfolders but excluding one subfolder

```
New-CAWindowsFSAuditObject -IncludePath "C:\ExampleDirectory" -IncludePathType
SystemFolder -IncludeScope ScopeSubTree -AuditEvents $auditEvents -IncludeMask "*"
-ExcludeFolderPaths "C:\ExampleDirectory\ExcludedDirectory"
```

Example: Monitoring a directory for one level for all file type except for .tmp files

```
New-CAWindowsFSAuditObject -IncludePath "C:\ExampleDirectory" -IncludePathType
SystemFolder -IncludeScope ScopeOneLevel -AuditEvents $auditEvents -IncludeMask "*"
-ExcludeFilePaths "*.tmp"
```

New-CAWindowsFSAuditTemplate

To enable Windows File System auditing, you must first create an auditing template for each file or folder to audit. Each auditing template defines the files or folders to audit, the auditing scope, and the excluded processes.

Use this command to create a Windows file system auditing template.

Table 51. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TemplateName	The template name.
-AuditObjects	The folder or file path objects created using New-CAWindowsFSAuditObject.
-ExcludeProcess (Optional)	The list of processes to exclude from auditing. The default is none.
-DiscardTooltipEvents (Optional)	Multiple folder open events are generated by tooltips (folder content information that is displayed when you hover your mouse over a folder) because Windows Explorer navigates the folder tree for all the sub-folders when you hover over the parent folder to see the tooltip. To ignore the folder opened events generated by this action set this parameter to 'true'.
-DiscardBrowsingEvents (Optional)	Multiple file open events are generated by file scans because Windows Explorer opens and reads the header of all files contained in an opened folder for information to display in the window. To ignore the file open events generated by this action set this parameter to 'true'.
-Disabled (Optional)	Specifies whether the template is enabled or disabled. Default is set to false.

Example: Create a Windows File System template

```
New-CAWindowsFSAuditTemplate -Connection $connection -TemplateName 'New-FSTemplate'
-AuditObjects $auditObject -ExcludeProcess $excludeProcess -DiscardTooltipEvents
$true -DiscardBrowsingEvents $true -Disabled $false
```

Remove-CAWindowsFSAuditTemplate

Use this command to delete a Windows File System auditing template.

Table 52. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The CAWindowsFSAuditTemplate object to remove. Obtain the template objects using the Get-CAWindowsFSAuditTemplates command and filter to select the object to remove.
-Force (Optional)	Removes template without prompting for a confirmation. The default is false.

Example: Remove a Windows File System template

```
Remove-CAWindowsFSAuditTemplate -Connection $connection -Template $removeTemplate
```

Set-CAWindowsFSAuditTemplate

Use this command to edit an existing Windows File System auditing template.

Table 53. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The CAWindowsFSAuditTemplate object to edit. Obtain the template objects using the Get-CAWindowsFSAuditTemplates command and filter to select the object to update.
-TemplateName (Optional)	The template name.
-AuditObjects (Optional)	The folder or file path objects created using New-CAWindowsFSAuditObject.
-ExcludeProcess (Optional)	The list of processes to exclude from auditing. The default is none.
-DiscardTooltipEvents (Optional)	Multiple folder open events are generated by tooltips (folder content information that is displayed when you hover your mouse over a folder) because Windows Explorer navigates the folder tree for all the sub-folders when you hover over the parent folder to see the tooltip. To ignore the folder opened events generated by this action set this parameter to 'true'.
-DiscardBrowsingEvents (Optional)	Multiple file open events are generated by file scans because Windows Explorer opens and reads the header of all files contained in an opened folder for information to display in the window. To ignore the file open events generated by this action set this parameter to 'true'.
-Disabled (Optional)	Set to true or false to enable or disable the template.

Example: Excluding and changing the template name

```
Set-CAWindowsFSAuditTemplate -Connection $connection -Template $Template -
ExcludeProcess "avsoftware.exe" -TemplateName "NewTemplateName"
```

Get-CAWindowsFSAuditTemplates

Use this command to see all the Windows File System auditing templates available within your installation.

Table 54. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get a list of all Windows File Server templates

```
Get-CAWindowsFSAuditTemplates -Connection $connection
```

Example: Get a template based on name

```
$template = Get-CAWindowsFSAuditTemplates -Connection $connection | where
TemplateName -eq TemplateName
```

Get-CAWindowsFSEventClassInfo

Use this command to get a list of all available Windows File System auditing event classes.

Table 55. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get a list of all Windows File Server event classes

```
Get-CAWindowsFSEventClassInfo -Connection $connection
```

Managing SQL Extended Events Auditing (Preview)

SQL Server Extended Events allow users to gather information on the performance of their SQL database. These commands allow you to create and manage SQL Extended Events auditing templates for auditing SQL Extended Events.

NOTE:

- A Change Auditor SQL Server license is required for SQL Extended Events auditing.
- Managing SQL Extended Events auditing templates is only available to Change Auditor administrators or users that hold a custom Change Auditor role that includes the View_SQL_Template operation.
- The specified SQL login account must have the following SQL Server permissions:
  - Alter any event session
  - View server state
  - Connect SQL
  - View any database

  Alternatively, the account must have an SQL Server role that contains these permissions, for example 'Sysadmin'.
- If using Windows authentication for the SQL Login account, and the specified agent for the auditing template is not installed on the target SQL server, the agent host computer account must also be added to the SQL Security Logins on the target SQL server with the same permissions requirement as the specified SQL login account for the auditing template.
- If using Windows authentication for the SQL Login account, and the agent for the auditing template is installed on the target SQL server, the local computer 'NT AUTHORITY\SYSTEM' account must be added to the SQL Security Logins on the target SQL server with the View Server State permission granted.

- Get-CASQLExtendedEventsInfo
- New-CASQLExtendedEventsFilter
- New-CASQLExtendedEventsObject
- New-CASQLExtendedEventsTemplate
- Get-CASQLExtendedEventsTemplates
- Remove-CASQLExtendedEventsTemplate

Get-CASQLExtendedEventsInfo

Use this command to retrieve the list of event names and filters available from the SQL server to use when configuring the SQL Extended Events template. Change Auditor audits event information from the Admin, Operational, and Analytic channels.

Table 56. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-SQLServerName	The name or IP of the SQL Server and the instance name if a named instance. For example, SQLTestServer\InstanceName.
-SQLServerLoginCredential	The SQL server credentials used to retrieve the list of available events and filters from the SQL server.

Example: Get all available SQL Extended Events event names and filters (predicates) available from the SQL Server.

```
Get-CASQLExtendedEventsInfo –Connection $connection -SQLServerName $sqlservername -
SQLServerLoginCredential $dbcredential
```

New-CASQLExtendedEventsFilter

Use this command to specify a filter for the SQL Extended Events to audit when creating templates.

Table 57. Parameter description

Parameter	Description
-EventsInfo	The available event and filter information obtained using the Get-CASQLExtendedEventsInfo command.
-FieldName	The field on which to filter.
-Operator	The operator to be used for comparison. See the output obtained from the Get-CASQLExtendedEventsInfo command for available operators for the specified filter field.
-Value	The value to be used for comparison.
-FilterType	The type of filter AND or OR.

Example: Filter on a specified field and value.

```
New-CASQLExtendedEventsFilter -EventsInfo $eventsInfo -FieldName database_name -
Operator Equals -Value testdb1 -FilterType 'AND'
```

New-CASQLExtendedEventsObject

Use this command to specify the SQL Extended Events to audit.

Table 58. Parameter description

Parameter	Description
-EventsInfo	The available event and filter information obtained using thee Get-CASQLExtendedEventsInfo command.
-EventNames	A string array of event names to be included.
-EventPackages	A string array of event packages to be applied for the specified array of event names when the object is created. The array values for parameters -EventNames and -EventPackages must be the same length as each index element of each array is paired with one another. You can specify an empty value for the default package.

Example: Populate a SQL Extended Events audited event name list

```
New-CASQLExtendedEventsObject -EventsInfo $sqlExtendedEventClasses -EventNames
"login_event","database_stopped","error_reported" -EventPackages
"sqlserver","sqlserver","xesvlpkg"
```

New-CASQLExtendedEventsTemplate

Use this command to create SQL Extended Events auditing templates.

Table 59. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-SQLServerName	The name or IP of the SQL Server and the instance name if a named instance. For example, SQLTestServer\InstanceName.
-SQLServerLogonCredential	The SQL server logon credential. **NOTE:** • Both SQL and Windows authentication are supported. • If using Windows authentication for the SQL Login account, and the specified agent for the auditing template is not installed on the target SQL server, the agent host computer account must also be added to the SQL Security Logins on the target SQL server with the same permissions requirement as the specified SQL login account for the auditing template. • If using Windows authentication for the SQL Login account, and the agent for the auditing template is installed on the target SQL server, the local computer 'NT AUTHORITY\SYSTEM' account must be added to the SQL Security Logins on the target SQL server with the View Server State permission granted.
-Name	A unique name for the template.
-ExtendedEvents	The list of events to audit using New-CASQLExtendedEventsObject.
-Filters (Optional)	A list of event filters using New-CASQLExtendedEventsFilter.
-MaxMemorySize (Optional)	SQL Extended Events maximum memory size in megabytes. Minimum is 250 MB (default if parameter not specified).
-Disabled (Optional)	Set to determine if the template is enabled or disabled. By default this is set to False.
-AgentInfo (Optional)	An agent object obtained using the Get-CAAgents command. If not specified, it will expect an agent installed on the SQL server to be audited. The agent is used for SQL Extended Events session management and event auditing.

Example: New SQL Extended Events template

```
New-CASQLExtendedEventsTemplate -Connection $connection -AgentInfo $Agent -
SQLServerName $sqlServerName -SQLServerLoginCredential $sqlCredential -Name
'testXEventTemplate' -ExtendedEvents $events -Filters $filters
```

Get-CASQLExtendedEventsTemplates

Use this command to see all the SQL Extended Events templates that have been created.

Table 60. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get all the SQL Extended Events templates

```
Get-CASqlExtendedEventsTemplates -Connection $connection
```

Example: Get SQL Extended Events templates filtered by a specified name

```
Get-CASqlExtendedEventsTemplates -Connection $connection | Filter.Where(_$.name =
"MyTemplate")
```

Remove-CASQLExtendedEventsTemplate

Use this command to delete a specified SQL Extended Events template.

Table 61. Parameter description

Parameter	Description
-Connection	A connection obtained by using Connect-CAClient.
-Template	The template object obtained using Get-CASQLExtendedEventsTemplates.

Example: Remove all the SQL Extended Events templates

```
Remove-CASQLExtendedEventsTemplate -Connection $connection -Template $template
```

Managing Microsoft Entra ID auditing

Change Auditor audits activity in the Microsoft Entra admin center that corresponds to the events in the Microsoft Entra ID auditing logs and sign-in activity. Managing Microsoft Entra ID auditing is available through the following PowerShell commands:

- New-CAAzureADTemplate
- Get-CAAzureADTemplates
- Set-CAAzureADTemplate

NOTE: When you delete a template (see Remove-CAAgentTemplate), the web application created in Microsoft Entra ID remains. You can delete the web application using the Microsoft Entra admin center. If you do not have the portal, see https://technet.microsoft.com/en-us/library/dn832618.aspx for instructions.

NOTE: If your organization uses a proxy server to connect to the internet, you must configure the agent settings to audit Microsoft Entra ID and Microsoft 365 targets. (See Set-CAConfiguration)

The following sample scripts are available in the Change Auditor client folder. By default they are located here: C:\Program Files\Quest\ChangeAuditor\Client\PowerShell Sample Scripts:

- CreateAzureADTemplate
- CreateAzureADTemplateUsingWebAppKey
- RemoveAzureADTemplate
- DisableAzureADTemplate
- ModifyAzureADTemplate-ChangeAgent
- GetAzureADTemplates

New-CAAzureADTemplate

Use this command to create a template for auditing Microsoft Entra ID.

Table 62. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent is used for Microsoft Entra ID auditing. **NOTE:** The agent must be allowed to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Microsoft Entra ID auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Microsoft Entra ID auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-CreateWebApp (Optional)	Specifies that you want to create a new web application. You will need to login to register Change Auditor in the tenant and ensure the required consent has been granted. Note: Internet access is required. The Microsoft sign-in page opens automatically. ***To grant permission for all administrators to create a web application:*** 1 Select an account with the Global Administrator role. 2 Enter the required password, and select **Sign in**. 3 Review the required permissions for the Change Auditor Configuration Assistant on the Microsoft Entra ID consent page. (Consent is only required once per tenant. You may, however, be prompted to enter your log on credentials when creating a new web application.) To apply the consent to all the users in your organization, click to enable **Consent on behalf of your organization** and click **Accept**. To apply the consent for just the current signed-in user simply click **Accept**. **NOTE:** When you specify this parameter a new web application is created and assigned to the template.
-DeploymentType	Specifies the tenant type (Commercial, GCC, or GCCHigh). If not set, the default is Commercial.
-Tenant	The Microsoft Entra tenant/directory that you want to audit (for example: yourTenantName.onmicrosoft.com).
-AuditLogs	Specifies whether or not to audit the Microsoft Entra ID audit logs. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-SignIns	Specifies whether or not to audit Microsoft Entra sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.

Parameter	Description
-HistoricalEventCollectionHours (Optional)	Specifies how many hours the agent should go back in time to start event collection. The parameter accepts values from 1 to 720. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionDays parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected.
-HistoricalEventCollectionDays (Optional)	Specifies how many days the agent should go back in time to start event collection. The parameter accepts values from 1 to 30. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionHours parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected.
-Disabled (Optional)	Specifies whether auditing is enabled or disabled for Microsoft Entra ID.

Example: Creating Microsoft Entra ID auditing template that will collect events generated 30 days in the past.

```
$connection = Connect-CAClient –InstallationName 'Default'

$agent = Get-CAAgents –Connection $connection | where{$_.agentfqdn -like
"CAAGENT.DOMAIN.COM"}  *Keep in Uppercase

New-CAAzureADTemplate -Connection $connection -CreateWebApp -Tenant $tenant
-AgentInfo $agent –HistoricalEventCollectionDays 30 -SignIns $True -AuditLogs $True
```

Create a template using an existing web application

Alternatively, use these parameters if you are using a pre-created web application that Change Auditor will use for authentication.

For details on integrating applications with Microsoft Entra ID and creating a web application, consult the Microsoft documentation. When creating a web application in the Microsoft Entra admin center, you are required to provide the following URLs: Sign-On URL, App ID URL. Specify any URL address that is unique to your tenant (for example: http://ChangeAuditorApp) for each of them.

The following permissions must be assigned to the web application:

Table 63. Required permission

System	Permissions
Office 365 Management APIs	Application Permissions: • ActivityFeed.Read – Application - Read activity data for your organization
Microsoft Graph	Application Permissions: • AuditLog.Read.All – Application - Read all audit log data • Directory.Read.All – Application - Read directory data • IdentityRiskEvent.Read.All – Application - Read all identity risk information

Once the required permissions are applied, click **Grant admin consent for**… and confirm with **Yes**.

Table 64. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent will be used for Microsoft Entra ID auditing. **NOTE:** The agent must be allowed to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Microsoft Entra ID auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Microsoft Entra ID auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-DeploymentType	Specifies the tenant type (Commercial, GCC, or GCCHigh). If not set, the default is Commercial.
-Tenant	The Microsoft Entra tenant/directory that you want to audit (for example: yourTenantName.onmicrosoft.com).
-AuditLogs	Specifies whether or not to audit the Microsoft Entra ID audit logs. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-SignIns	Specifies whether or not to audit Microsoft Entra sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-WebAppId	A web application ID. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant. **NOTE:** Microsoft Entra ID and Microsoft 365 must each have their own dedicated web application.
-WebAppKey	The key assigned to the web application specified for the WebAppId parameter. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant. **NOTE:** When using this parameter, you cannot also specify –WebAppCreationCredential parameter.
-HistoricalEventCollectionHours (Optional)	Specifies how many hours the agent should go back in time to start event collection. The parameter accepts values from 1 to 720. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionDays parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected.
-HistoricalEventCollectionDays (Optional)	Specifies how many days the agent should go back in time to start event collection. The parameter accepts values from 1 to 30. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionHours parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected.
-Disabled (Optional)	Specifies whether auditing is enabled or disabled for Microsoft Entra ID.

Example: Creating an Microsoft Entra ID auditing template using a pre-created web application

that will collect events generated 30 days in the past.

```
New-CAAzureADTemplate -Connection $connection -AgentInfo $agent -WebAppKey
$webAppKey -WebAppId $webAppId -Tenant $tenant –HistoricalEventCollectionDays 30
-SignIns $True -AuditLogs $True
```

Set-CAAzureADTemplate

Use this command to edit the web application key and ID, and the agent in an existing Microsoft Entra ID template. This also allows you to replace an expired or revoked web application.

NOTE:

- You cannot edit the type of activity to audit (audit logs and/or sign-ins) and the WebAppId, WebApp Key, and agent at the same time. Activity must be edited in a separate command.

Table 65. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent will be used for Microsoft Entra ID auditing. **NOTE:** The agent must be allowed to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Microsoft Entra ID auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Microsoft Entra ID auditing. This is the port that is used for communicating with the tenant. **NOTE:** The web application ID and key values are encrypted; therefore, each time you change the agent associated with a template you must explicitly specify the values again.
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	A template object obtained by the Get-CAAzureADTemplates command.
-CreateWebApp (Optional)	Specifies that you want to create a new web application. You will need to login to register Change Auditor in the tenant and ensure the required consent has been granted. Note: Internet access is required. The Microsoft sign-in page opens automatically. ***To grant permission for all administrators to create a web application:*** 1 Select an account with the Global Administrator role. 2 Enter the required password, and select **Sign in**. 3 Review the required permissions for the Change Auditor Configuration Assistant on the Microsoft consent page. (Consent is only required once per tenant. You may, however, be prompted to enter your log on credentials when creating a new web application.) To apply the consent to all the users in your organization, click to enable **Consent on behalf of your organization** and click **Accept**. To apply the consent for just the current signed-in user simply click **Accept**. **NOTE:** When you specify this parameter a new web application is created and assigned to the template.
-AuditLogs	Specifies whether or not to audit the Microsoft Entra ID audit logs. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-SignIns	Specifies whether or not to audit Microsoft Entra sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.

Parameter	Description
-WebAppId	A web application ID. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant. **NOTE:** Microsoft Entra ID and Microsoft 365 must each have their own dedicated web application.
-WebAppKey	The key assigned to the web application specified for the WebAppId parameter. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant.

Example: Modify web application credentials in an auditing template

```
Set-CAAzureADTemplate -Connection $connection -Template $template -WebAppKey
$webAppKey -WebAppId $webAppId
```

Example: Add auditing of all activities to an existing template

```
Set-CAAzureADTemplate -Connection $connection -Template $template -SignIns $True
-AuditLogs $True
```

Get-CAAzureADTemplates

Use this command to see all the Microsoft Entra ID templates available within your installation.

Table 66. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get a list of all Microsoft Entra ID templates

```
Get-CAAzureADTemplates -Connection $connection
```

Managing Office 365 auditing

Change Auditor for Exchange and Change Auditor for SharePoint have been extended to include the auditing of activities taking place in Exchange Online, SharePoint Online, and OneDrive for Business. The following commands are available to manage Office 365 auditing:

- New-CAO365Template
- Set-CAO365Template
- Get-CAO365Templates
- Remove-CAO365Template
- Get-CAO365ExchangeMailboxes
- Add-CAO365ExchangeTemplateMailboxes
- Remove-CAO365ExchangeTemplateMailboxes
- Get-CAO365ExchangeTemplateMailboxes

NOTE: When you delete a template (see Remove-CAAgentTemplate), the web application created in Microsoft Entra ID remains. You can delete the web application using the Microsoft Entra admin center. If you do not have the portal, see https://technet.microsoft.com/en-us/library/dn832618.aspx for instructions.

NOTE: If your organization uses a proxy server to connect to the internet, you must configure the agent settings to audit Microsoft Entra ID and Microsoft 365 targets. (See Set-CAConfiguration)

New-CAO365Template

Use this command to create a template for auditing Office 365 Exchange Online, SharePoint Online, and OneDrive for Business.

Table 67. Available parameters

Parameter	Description
-AgentInfo	An agent obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Office 365 auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Office 365 auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-CreateWebApp (Optional)	Specifies that you want to create a new web application. You will need to login to register Change Auditor in the tenant and ensure the required consent has been granted. Note: Internet access is required. The Microsoft sign-in page opens automatically. ***To grant permission for all administrators to create a web application:*** 1 Provide an account with the Global Administrator role. 2 Enter the required password, and select **Sign in**. 3 Review the required permissions for the Change Auditor Configuration Assistant on the Microsoft consent page. (Consent is only required once per tenant. You may, however, be prompted to enter your log on credentials when creating a new web application.) To apply the consent to all the users in your organization, click to enable **Consent on behalf of your organization** and click **Accept**. To apply the consent for just the current signed-in user simply click **Accept**. **NOTE:** When you specify this parameter a new web application is created and assigned to the template.
-DeploymentType	Specifies the tenant type (Commercial, GCC, or GCCHigh). If not set, the default is Commercial.
-Tenant	The Microsoft Entra tenant/directory that you want to audit (for example: yourTenantName.onmicrosoft.com).
-AuditAdministration (Optional)	Specifies whether to audit administration events.
-AuditOrganization (Optional)	Specifies whether to audit all Exchange Online mailboxes accessed by users other than the mailbox owner.

Parameter	Description
-CertificateFile (Optional)	The filename of an exported X509 certificate with private key. **NOTE:** This parameter is not required if the -CertificateThumbprint or -GenerateCertificate parameter is specified.
-CertificateFilePassword (Optional)	The password for the certificate file. **NOTE:** • If the -CertificateFile parameter is not specified this parameter is ignored. • If the -CertificateFile parameter is specified this parameter is required. The private key must be protected by password. Protection with security principal is not supported.
-CertificateThumbprint (Optional)	The thumbprint of a certificate that is located in the user's personal certificate store on the host workstation and must have a private key (string format). **NOTE:** • The private key must be marked as exportable. • This parameter is not required if the -CertificateFile or -GenerateCertificate parameter is specified.
-GenerateCertificate (Optional)	If specified, will generate a new self-signed certificate.
-Disabled (Optional)	Specifies whether the auditing template is enabled or disabled.
-EnableExchangeOnline (Optional)	Specifies whether Exchange Online auditing is enabled or disabled.
-EnableOneDrive (Optional)	Specifies whether OneDrive for Business auditing is enabled or disabled.
-EnableSharePoint (Optional)	Specifies whether SharePoint Online auditing is enabled or disabled.
-HistoricalEventCollectionHours (Optional)	Specifies how many hours the agent should go back in time to start event collection. The parameter accepts values from 1 to 168. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionDays parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected. **NOTE:** The historical data returned is based on the information in the Office 365 audit logs. This may not reflect the configuration options in the current template.
-HistoricalEventCollectionDays (Optional)	Specifies how many days the agent should go back in time to start event collection. The parameter accepts values from 1 to 7. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionHours parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected. **NOTE:** The historical data returned is based on the information in the Office 365 audit logs. This may not reflect the configuration options in the current template.
-ExcludedOperations (Optional)	String that specifies events to exclude from the Office 365 OneDrive for Business event, Office 365 SharePoint Online event, and Office 365 Exchange Online event. These generic dynamically constructed events are created when associated activity is detected that does not have a corresponding event defined in Change Auditor.
-OverwriteTenantMailboxAuditing (Optional)	Specifies whether the template auditing settings will overwrite the existing tenant auditing settings.

Example: Create a template that audits both Exchange Online administration and mailbox non-owner events and will collect events generated 7 days in the past.

```
New-CAO365Template -Connection $connection -Tenant $tenant -AgentInfo $agent -
CreateWebApp -GenerateCertificate -EnableExchangeOnline $true -AuditAdministration
$true -AuditOrganization $true -HistoricalEventCollectionDays 7
```

Create a template using an existing web application

When you create or edit an Office 365 auditing template and you select to use an existing web application, it must be configured to support certificate authentication. See the Microsoft Entra ID and Microsoft 365 User Guide for the required steps.

For more details on integrating applications with Microsoft Entra ID and creating a web application, consult the Microsoft documentation. When creating a web application in the Microsoft Entra admin center, you are required to provide the following URLs: Sign-On URL, App ID URL. Specify any URL address that is unique to your tenant (for example: http://ChangeAuditorApp) for each of them.

Table 68. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Office 365 auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Office 365 auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-DeploymentType	Specifies the tenant type (Commercial, GCC, or GCCHigh). If not set, the default is Commercial.
-Tenant	The Microsoft Entra tenant/Directory that you would like Change Auditor to audit (for example: yourTenantName.onmicrosoft.com).
-WebAppId	A web application Id. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant. **NOTE:** Microsoft Entra ID and Microsoft 365 must each have their own dedicated web application. **NOTE:** When using this parameter, you cannot also specify the –CreateWebApp parameter.
-WebAppKey	The key assigned to the web application specified for the WebAppId parameter. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant.
-AuditAdministration (Optional)	Specifies whether to audit administration events.
-CertificateFile (Optional)	The filename of an exported X509 certificate with private key. **NOTE:** This parameter is not required if the -CertificateThumbprint or -GenerateCertificate parameter is specified.
-CertificateFilePassword (Optional)	The password for the certificate file. **NOTE:** • If the -CertificateFile parameter is not specified this parameter is ignored. • If the -CertificateFile parameter is specified this parameter is required. The private key must be protected by password. Protection with security principal is not supported.
-CertificateThumbprint (Optional)	The thumbprint of a certificate that is located in the user's personal certificate store on the host workstation and must have a private key (string format). **NOTE:** • The private key must be marked as exportable. • This parameter is not required if the -CertificateFile or -GenerateCertificate parameter is specified.
-AuditOrganization (Optional)	Specifies whether to audit all Exchange Online mailboxes accessed by users other than the mailbox owner.
-Disabled (Optional)	Specifies whether the auditing template is enabled or disabled.
-EnableExchangeOnline (Optional)	Specifies whether Exchange Online auditing is enabled or disabled.

Parameter	Description
-EnableOneDrive (Optional)	Specifies whether OneDrive for Business auditing is enabled or disabled.
-EnableSharePoint (Optional)	Specifies whether SharePoint Online auditing is enabled or disabled.
-HistoricalEventCollectionDays (Optional)	Specifies how many days the agent should go back in time to start event collection. The parameter accepts values from 1 to 7. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionHours parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected. **NOTE:** The historical data returned is based on the information in the Office 365 audit logs. This may not reflect the configuration options in the current template.
-HistoricalEventCollectionHours (Optional)	Specifies how many hours the agent should go back in time to start event collection. The parameter accepts values from 1 to 168. **NOTE:** When using this parameter, you cannot also specify the –HistoricalEventCollectionDays parameter. **NOTE:** Using this parameter may cause a duplication of events if the same events have been previously collected. **NOTE:** The historical data returned is based on the information in the Office 365 audit logs. This may not reflect the configuration options in the current template.
-ExcludedOperations (Optional)	String that specifies events to exclude from the Office 365 OneDrive for Business event, Office 365 SharePoint Online event, and Office 365 Exchange Online event. These generic dynamically constructed events are created when associated activity is detected that does not have a corresponding event defined in Change Auditor.
-OverwriteTenantMailboxAuditing (Optional)	Specifies whether the template auditing settings will overwrite the existing tenant auditing settings.

Example: Create a template that audits both Exchange Online administration and mailbox non-owner events and will collect events generated 7 days in the past.

```
New-CAO365Template -Connection $connection -Tenant $tenant -AgentInfo $agent -
WebAppId $webAppID -WebAppKey $webAppKey1 -CertificateFile
'C:\Users\user.domain\Desktop\CertificateFile.pfx' -CertificateFilePassword
$password -EnableExchangeOnline $true -AuditAdministration $true -AuditOrganization
$true -HistoricalEventCollectionDays 7
```

Set-CAO365Template

Use this command to edit the account used to access Office 365 Exchange Online, the type of service and events to audit, and select a new agent.

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	A template object obtained by using the Get-CAO365Templates command.

Parameter	Description
-CreateWebApp (Optional)	Specifies that you want to create a new web application. You will need to login to register Change Auditor in the tenant and ensure the required consent has been granted. Note: Internet access is required. The Microsoft sign-in page opens automatically. ***To grant permission for all administrators to create a web application:*** 1 Select an account with the Global Administrator role. 2 Enter the required password, and select **Sign in**. 3 Review the required permissions for the Change Auditor Configuration Assistant on the Microsoft consent page. (Consent is only required once per tenant. You may, however, be prompted to enter your log on credentials when creating a new web application.) To apply the consent to all the users in your organization, click to enable **Consent on behalf of your organization** and click **Accept**. To apply the consent for just the current signed-in user simply click **Accept**. **NOTE:** When you specify this parameter a new web application is created and assigned to the template.
-WebAppId	A web application Id. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant. **NOTE:** Microsoft Entra ID and Microsoft 365 must each have their own dedicated web application. **NOTE:** When using this parameter, you cannot also specify the –CreateWebApp parameter.
-WebAppKey	The key assigned to the web application specified for the WebAppId parameter. This application is needed for Change Auditor to authenticate to your Microsoft Entra tenant.
-AgentInfo (Optional) **NOTE:** This parameter is required if you are modifying the web application.	An agent object obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Microsoft Entra ID. • If the agent is separated from the coordinator by a firewall, you must create a firewall exception for port 8373 on every agent computer to be used for Office 365 auditing. This is the default port that enables the coordinator to communicate with the agent. A different port number can, however, be specified by running the Set-CAConfiguration command. • A firewall outbound exception for remote port 443 (https) must exist for every agent computer that will be used for Office 365 auditing. This is the port that is used for communicating with the tenant. **NOTE:** The web application ID and key values are encrypted; therefore, each time you change the agent associated with a template you must explicitly specify the values again.
-AuditAdministration (Optional)	Specifies whether to audit administration events.
-AuditOrganization (Optional)	Specifies whether to audit all Exchange Online mailboxes accessed by non-owners.

Parameter	Description
-CertificateFile (Optional)	The filename of an exported X509 certificate with private key. **NOTE:** This parameter is not required if the -CertificateThumbprint or -GenerateCertificate parameter is specified, or if the web application is not being modified.
-CertificateFilePassword (Optional)	The password for the certificate file. **NOTE:** • If the -CertificateFile parameter is not specified this parameter is ignored. • If the -CertificateFile parameter is specified this parameter is required. The private key must be protected by password. Protection with security principal is not supported.
-CertificateThumbprint (Optional)	The thumbprint of a certificate that is located in the user's personal certificate store on the host workstation and must have a private key (string format). **NOTE:** • The key must be marked as exportable. • This parameter is not required if the -CertificateFile or -GenerateCertificate parameter is specified, or if the web application is not being modified.
-GenerateCertificate (Optional)	If specified, will generate a new self-signed certificate.
-EnableExchangeOnline (Optional)	Specifies whether Exchange Online auditing is enabled or disabled.
-EnableOneDrive (Optional)	Specifies whether OneDrive for Business auditing is enabled or disabled.
-EnableSharePoint (Optional)	Specifies whether SharePoint Online auditing is enabled or disabled.
-ExcludedOperations (Optional)	String that specifies events to exclude from the Office 365 OneDrive for Business event, Office 365 SharePoint Online event, and Office 365 Exchange Online event. These generic dynamically constructed events are created when associated activity is detected that does not have a corresponding event defined in Change Auditor.
-OverwriteTenantMailboxAuditing (Optional)	Specifies whether the template auditing settings will overwrite the existing tenant auditing settings.

Example: Enable auditing all Office 365 Exchange Online mailboxes accessed by non-owners

```
Set-CAO365Template -Connection $connection -Template $template
-AuditOrganization $true
```

Example: Enable auditing of SharePoint Online and OneDrive for Business

```
Set-CAO365Template -Connection $connection -Template $template -EnableSharePoint
$true -EnableOneDrive $true
```

Example: Generate a new web application and new certificate for an existing O365 auditing template.

```
Set-CAO365Template -Connection $connection -Template $template -CreateWebApp -
GenerateCertificate -AgentInfo $agent
```

Example: Replace the web application

```
Set-CAO365Template -Connection $connection -Template $template -WebAppId $webAppId -
WebAppKey $webAppKey -CertificateThumbprint $certificateThumbprint -AgentInfo $agent
```

NOTE: The certificate with the exportable private key is in the current client user personal certificate store and the web application is updated with the specified certificate (.cer file is uploaded to the web app - matching specified thumbprint).

Example: Replace the agent

```
Set-CAO365Template -Connection $connection -Template $template -AgentInfo $agent
```

Get-CAO365Templates

Use this command to see all the Office 365 templates available within your installation.

Table 69. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Get a list of all Office 365 templates

```
Get-CAO365Templates -Connection $connection
```

Remove-CAO365Template

Use this command to remove a template for auditing Office 365 Exchange Online, SharePoint Online, and OneDrive for Business.

Table 70. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Tenant	The Office 365 tenant that is used for auditing. For example, yourTenantName.onmicrosoft.com.

Example: Remove an Office 365 template

```
Remove-CAO365Template -Connection $connection -Tenant $tenant
```

Get-CAO365ExchangeMailboxes

Use this command to find specific mailboxes that can be added to an existing Office 365 Exchange Online template.

NOTE: To run this command, you must first create an Office 365 auditing template. See New-CAO365Template.

Table 71. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Tenant	The Office 365 tenant that is used for auditing. For example, yourTenantName.onmicrosoft.com.
-SearchText (Optional)	The search criteria specified as the mailbox display name. This can be the full name of the mailbox to return a specific mailbox or the starting characters to return a list of mailboxes that start with those characters.
-Skip (Optional)	The number of objects to exclude from the list of returned objects, starting from the top.
-First (Optional)	The number of objects to return.
-IncludeTotalCount (Optional)	The total number of objects in the data set. Values specified for the First or Skip parameters do not impact this count.

Example: Find all Office 365 mailboxes that start with the letter a

```
Get-CAO365ExchangeMailboxes -Connection $connection -Tenant $tenant -SearchText "a"
```

Add-CAO365ExchangeTemplateMailboxes

Use this command to audit specific mailboxes in your organization by adding them to an existing Office 365 Exchange Online template.

Table 72. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	A template object obtained by using the Get-CAO365Templates command.
-Mailboxes	Mailbox objects obtained by using the Get-CAO365ExchangeMailboxes command.
-AuditOwnerEvents (Optional)	A switch that indicates that the added mailboxes will be audited for owner activity in addition to the non-owner activity. By default, the mailboxes will be audited for non-owner mailbox activity only. **IMPORTANT:** It is recommended that you select owner auditing for critical mailboxes only. Owner auditing for a large number of mailboxes produces many events that may affect performance.
-OverwriteExisting (Optional)	If the mailboxes already exist in the template, this switch indicates that the mailboxes will have their current owner/non-owner auditing settings overwritten with new settings.

Example: Add Office 365 mailboxes to the existing Exchange Online template

```
Add-CAO365ExchangeTemplateMailboxes -Connection $connection -Template $template -
Mailboxes $mailboxes –AuditOwnerEvents
```

Remove-CAO365ExchangeTemplateMailboxes

Use this command to remove mailboxes from an existing Office 365 Exchange Online template.

Table 73. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	A template object obtained by using the Get-CAO365Templates command.
-Mailboxes	Mailbox objects obtained by using the Get-CAO365ExchangeMailboxes command.
-All (Optional)	A switch that indicates that all mailboxes will be removed from the template.

Example: Remove all Office 365 mailboxes from the existing Exchange Online template

```
Remove-CAO365ExchangeTemplateMailboxes -Connection $connection -Template $template –
All
```

Get-CAO365ExchangeTemplateMailboxes

Use this command to retrieve a list of mailboxes being audited by a particular Office 365 Exchange Online template.

Table 74. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	A template object obtained by using the Get-CAO365Templates command.
-AuditTypeFilter	Parameter that allows you to narrow the search based on the type of activities being audited: non-owner only, owner (non-owner, owner), or any (non-owner only, owner and non-owner).
-DisplayNameFilter	The search criteria specified as the mailbox display name. This can be the full name of the mailbox to return a specific mailbox or the starting characters to return a list of mailboxes that start with those characters.
-Skip (Optional)	The number of objects to exclude from the list of returned objects, starting from the top.
-First (Optional)	The number of objects to return.
-IncludeTotalCount (Optional)	The total number of objects in the data set. Values specified for the First or Skip parameters do not impact this count.

Example: Get all Office 365 audited mailboxes from the existing Exchange Online template

```
Get-CAO365ExchangeTemplateMailboxes -Connection $connection -Template $template
```

Example: This example will return mailboxes that are not enabled for owner auditing where the display name starts with “Sam S”

```
Get-CAO365ExchangeTemplateMailboxes -Connection $connection -Template $template -
DisplayNameFilter "Sam S" -AuditTypeFilter NonOwnerOnly
```

Configuring a Quest On Demand Audit integration

Quest On Demand Audit is a Software as a Service (SaaS) application, available through quest-on-demand.com that provides extensive, customizable auditing of critical activities and detailed alerts about vital changes taking place in Microsoft 365 and Microsoft Entra ID.

On Demand Audit can also provide a single view of activity across hybrid Microsoft environments. By sending Change Auditor Active Directory event data, you can gain visibility to on premises changes (including events gathered up to 30 days prior to installing or upgrading Change Auditor).

To begin, you need to configure a connection between Change Auditor and your organization in On Demand Audit. Once the connection is made, On Demand Audit creates the required subscription used to send events from Change Auditor to On Demand Audit. For details on how Change Auditor uses subscriptions to send events, see the Change Auditor SIEM Integration Guide.

NOTE: Although the configuration is made through Change Auditor, the subscription is managed (edited and removed) through On Demand Audit. See the On Demand Audit User Guide for details.

- New-CAODAConfiguration
- Get-CAODAConfiguration

- Set-CAODAConfiguration

New-CAODAConfiguration

Use this command to create the connection required to send Change Auditor event data to On Demand Audit. When you run this command, you are presented with a dialog where you need to enter the information required to configure the connection. Enter your Quest account credentials to sign in to On Demand Audit and if prompted select the organization. By default, the current installation is used for the configuration name. If required, you can enter a different name for the configuration. This is the configuration name used in On Demand Audit; it does not change the Chane Auditor installation name.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.

Example: Create a subscription to send Active Directory event data to On Demand Audit

```
New-CAODAConfiguration -Connection $connection
```

Get-CAODAConfiguration

Use this command to see the details of the current On Demand Audit configuration.

Table 75. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-SubscriptionId (optional)	The ID of an existing On Demand Audit subscription.

Example: Get information about the On Demand Audit configuration

```
Get-CAODAConfiguration -Connection $connection
```

Command output

The command returns the following information.

Table 76. Available information about the subscription created by the configuration

Setting	Description
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
BatchesSent	Number of batches sent.
Enabled	Whether the subscription is enabled.
EventsSent	Number of events sent.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator that sent events.
LastEventResponse	The last event response. Provides the response in JSON format from the event receiver.

Table 76. Available information about the subscription created by the configuration

Setting	Description
LastEventTimeUTC	When the last event was sent.
NotificationInterval	How often how often (in milliseconds) notifications are sent.
StartTimeUTC	Starting point in time for events being sent.
Subscription Id	The subscription ID.
Subsystems	Subsystems that contain the event data being sent.
Webhook Subscription Id	The webhook subscription ID.

Set-CAODAConfiguration

Use this command to modify an On Demand Audit configuration.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. **NOTE:** The list order does not determine which coordinator is selected to send events.

Example: Set the allowed coordinators for the On Demand Audit configuration to the computers named "coordinator1" and "coordinator2"

```
Set-CAODAConfiguration -Connection $connection -AllowedCoordinators
@("coordinator1", "coordinator2")
```

Working with Active Directory protection templates

Enabling Active Directory protection allows you to lock down critical objects and attributes to prevent accidental or unauthorized creations, modifications, or deletions.

IMPORTANT: The following commands are only supported for protection templates that are stored in SQL. They will not function if the protection templates are stored in Active Directory.

The following commands are available to manage Active Directory protection:

- New-CAADProtectionTemplate
- New-CAProtectedObject
- Remove-CAProtectedObject
- New-CAForestCredential
- New-CAScheduledTimeRange
- Get-CAADProtectionTemplates
- Remove-CAADProtectionTemplate
- Set-CAADProtectionTemplate

New-CAADProtectionTemplate

Use this command to create an Active Directory protection template.

Table 77. Available parameters

Parameter	Description
-Credential	Credentials used to access the foreign forest.
-Name	The template name.
-ProtectedObjects	List of ProtectedObjects. See New-CAProtectedObject for details.
-Attributes (Optional)	List of attributes to protect. When AttributeType is not set to “All” this specifies the attributes for the template. Default is none.
-AttributeType (Optional)	This is applied to the list of attributes specified in the Attributes parameter. Possible values include “All”, “Only” and “AllExcept”. Default is All.
-OverrideAccounts (Optional)	Accounts allowed or not allowed to change the protected objects.
-OverrideAccountsDenied (Optional)	Specifies if you want to deny the list of user in the OverrideAccounts access. You can specify either $true or $false. Default is false which means that the user accounts are not denied access.
-AdminAccounts (Optional)	Accounts that can manage the protection template. Default is none.
-Locations (Optional)	IP addresses to protect. Default is none.
-LocationProtectionType (Optional)	Applied to the IP addresses specified by the Locations parameter. The potential values include ProtectAllLocations, ProtectSelectLocations, AllowSelectLocations, or ProtectUnknownLocations. Default is ProtectAllLocations.
-Schedule (Optional)	It is a list of PSCAScheduledTimeRange objects, created with the New-CAScheduledTimeRange cmdlet. Default is no specified schedule, which means that protection is always enabled. See New-CAScheduledTimeRange for details.

Example: Create an Active Directory protection template

```
$protectedObject = New-CAProtectedObject -ObjectDistinguishName “ObjectName” -
ProtectedScope ScopeObject -Operations Create

New-CAADProtectionTemplate -Connection $connection -Name TemplateSample1 -
ProtectedObjects $protectedObject
```

Example: Creating an Active Directory Protection template to protect objects in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

New-CAADProtectionTemplate -Connection $connection -Name $templateName -
ProtectedObjects $protectedObject -OverrideAccounts $overrideAccountDn -
AdminAccounts $adminAccountDn -Schedule $schedule -Credential $forestCredential
```

New-CAProtectedObject

Use this command to create a protected object to include in a protection template.

Table 78. Available parameters

Parameter	Description
-ObjectDistinguishName	Distinguish name of object to protect.
-ProtectedScope (Optional)	Scope of coverage for the protected object. Specify the scope using one of the following values: • ScopeObject • ScopeOneLevel • ScopeSubtree **NOTE:** The -ProtectedScope parameter is required for Active Directory protection templates.
-Operations	Operations to be denied for the selected object: • None • Create • Modify • Delete • Move • Link **NOTE:** You can specify multiple operations.

Example: Create a new protected object

```
New-CAProtectedObject -ObjectDistinguishName "ObjectName" -ProtectedScope
ScopeObject -Operations Create
```

Remove-CAProtectedObject

Use this command to remove protected objects from a protection template.

NOTE: Both the -ProtectedObject and the -All parameters are optional, however, one or the other must be specified in the command.

Table 79. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The PSCAProtectionTemplate object to remove protected objects from. Obtain the template objects using the Get-CAADProtectionTemplates command and filter to select the template object to remove protected objects from.
-Credential	Credentials used to access the foreign forest. **NOTE:** If the Active Directory protection template contains protected objects from a foreign forest, you must specify the -Credential parameter and value when getting the template object and when removing the protected object. The -Credential parameter value is obtained by running the New-CAForestCredential cmdlet.
-ProtectedObject (Optional)	Protected object (distinguished name).
-All (Optional)	Remove all the protected objects.

Example: Remove protected object

```
Remove-CAProtectedObject -Connection $connection -Template $template -
ProtectedObject $protectedObjectDn
```

Example: Remove protected object from a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

$templates = Get-CAADProtectionTemplates -Connection $connection -Credential
$forestCredential

Remove-CAProtectedObject -Connection $connection -Templates $template[2] -
ProtectedObject $protectedObjectDn -Credential $forestCredential
```

New-CAForestCredential

Use this command to input credentials for foreign forests when creating Active Directory protection templates with PowerShell.

Table 80. Available parameters

Parameter	Description
-ForestName	The name of the forest to access.
-Credential	Credentials used to access the foreign forest. The credential object is obtained by using the Get-Credential command.

Example: Creating an Active Directory Protection template to protect objects in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

New-CAADProtectionTemplate -Connection $connection -Name $templateName -
ProtectedObjects $protectedObject -OverrideAccounts $overrideAccountDn -
AdminAccounts $adminAccountDn -Schedule $schedule -Credential $forestCredential
```

New-CAScheduledTimeRange

Use this command to schedule when to enforce the protection.

Table 81. Available parameters

Parameter	Description
-Day	Spelled out day of the week to begin the protection. For example, Monday.
-StartTime	The time to start the protection. This parameter requires an integer and validates that the input is between 0 and 24 inclusive. This implies an hour of the day to start on.
-EndTime	The time to end the protection. This parameter requires an integer and validates that the input is between 0 and 24 inclusive. This implies an hour of the day to end on.

Example: Create a scheduled time range for a protected template

```
New-CAScheduledTimeRange -Day Monday -StartTime 7 -EndTime 18
```

Get-CAADProtectionTemplates

Use this command to see all the Active Directory protection templates that have been created including those in a foreign forest.

Table 82. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Credential	Credentials used to access the foreign forest.

Example: Get a list of all Active Directory Protection templates

```
Get-CAADProtectionTemplates -Connection $connection
```

Example: Get a list of all Active Directory Protection templates in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

Get-CAADProtectionTemplates -Connection $connection -Credential $forestCredential
```

Remove-CAADProtectionTemplate

Use this command to remove an Active Directory protection template.

Table 83. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Credential	Credentials used to access the foreign forest.
-Template	The PSCAProtectionTemplate object to remove. Obtain the template objects using the Get-CAADProtectionTemplates command and filter to select the object to remove.
-Force	Removes the template without providing confirmation.

Example: Remove an Active Directory protection template

```
Remove-CAADProtectionTemplate -Connection $connection -Template $template
```

Example: Remove an Active Directory Protection template in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

Remove-CAADProtectionTemplate -Connection $connection -Template $selectedTemplate -
Credential $forestCredential
```

Set-CAADProtectionTemplate

Use this command to modify Active Directory protection templates.

Table 84. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The PSCAProtectionTemplate object to update. Obtain the template objects using the Get-CAADProtectionTemplates command and filter to select the template object to update.
-TemplateName (Optional)	Sets the template name (string).
-Credential (Optional)	Credentials used to access the foreign forest. **NOTE:** If the Active Directory protection template contains protected objects and accounts from a foreign forest, you must specify the -Credential parameter and value when getting the template and when modifying the template object. The -Credential parameter value is obtained by running the New-CAForestCredential cmdlet.
-ProtectedObjects (Optional)	List of ProtectedObjects. See New-CAProtectedObject for details.
-Attributes (Optional)	List of attributes to protect. When AttributeType is not set to “All” this specifies the attributes for the template. Default is none specified.
-AttributeType (Optional)	This is applied to the list of attributes specified in the Attributes parameter. Possible values include “All”, “Only” and “AllExcept”. Default is All.
-OverrideAccounts (Optional)	Accounts allowed or not allowed to change the protected objects. String array of distinguished names.
-OverrideAccountsDenied (Optional)	Specifies if you want to deny the list of user in the OverrideAccounts access. You can specify either $true or $false. Default is false which means that the user accounts are not denied access.
-AdminAccounts (Optional)	Accounts that can manage the protection template. (If accounts are specified, then only those specified accounts can manage the template. If no accounts are specified, then all Change Auditor administrators can manage the template.) Default is none specified. This is a string array of distinguished names.
-Locations (Optional)	IP addresses to protect. Default is none specified.
-LocationProtectionType (Optional)	Applied to the IP addresses specified by the Locations parameter. The potential values include ProtectAllLocations, ProtectSelectLocations, AllowSelectLocations, or ProtectUnknownLocations. Default is ProtectAllLocations.
-Schedule (Optional)	It is a list of PSCAScheduledTimeRange objects, created with the New-CAScheduledTimeRange cmdlet. Default is no specified schedule, which means that protection is always enabled. See New-CAScheduledTimeRange for details.
-Disabled (Optional)	Specifies whether the template is enabled or disabled using the Boolean $true or $false.

Example: Modify a protection template

```
Set-CAADProtectionTemplate -Connection $connection -Template $template[2] -
ProtectedObjects $protectedObject1, $protectedObject2 -AdminAccounts $adminAccountDn
-Schedule $schedule -Disabled $False
```

Example: Modify a template that contains foreign forest objects

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

$templates = Get-CAADProtectionTemplates -Connection $connection -Credential
$forestCredential

Set-CAADProtectionTemplate -Connection $connection -Template $templates[2] -Schedule
$schedule -Credential $forestCredential
```

Working with GPO protection templates

Enabling GPO protection, allows you to prevent all changes to Group Policy Objects, regardless of the tool that is used to make the change. Protection includes both portions of the Group Policy data: the Group Policy Object (GPO) in Active Directory and the actual configuration data stored in the SYSVOL share on domain controllers

The following commands are available to manage GPO protection:

- New-CAGPOProtectionTemplate
- Get-CAGPOProtectionTemplates
- Set-CAGPOProtectionTemplate
- Remove-CAGPOProtectionTemplate

New-CAGPOProtectionTemplate

Use this command to create a GPO protection template.

Table 85. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Credential (Optional)	Credentials used to access the foreign forest.
-TemplateName	The template name.
-ProtectedObjects	List of ProtectedObjects. See New-CAProtectedObject for details.
-DoNotProtectWorkingCopies (Optional)	When enabled, GPOADmin working copies selected for the protection template (or in the AD forest if Enterprise is selected), are ignored by the template. The parameter accepts Boolean $true or $false. **NOTE:** Name-matching is used to identify GPOADmin temporary working copies based on a name prefix "[GPOADmin Working Copy] - ". The service account for the GPOADmin service should also be excluded from protection as an allowed account when the option to exclude GPOADmin working copies is selected.
-OverrideAccounts (Optional)	Accounts allowed or not allowed to change the protected objects.
-OverrideAccountsDenied (Optional)	Specifies if you want to deny the list of user in the OverrideAccounts access. You can specify either $true or $false. Default is false which means that the user accounts are not denied access.
-AdminAccounts (Optional)	Accounts that can manage the protection template. Default is none.
-Disabled (Optional)	Specifies whether the template is enabled or disabled using the Boolean $true or $false.

Example: Create a GPO Protection template

```
$ProtectedObjects = New-CAProtectedObject  -ObjectDistinguishName
"distinguishedName" -Operations Modify

New-CAGPOProtectionTemplate -Connection $connection -TemplateName TemplateSample1 -
ProtectedObjects $protectedObjects

$EnterpriseProtectedObject= New-CAProtectedObject -ObjectDistinguishName
"Enterprise" -Operations Modify

New-CAGPOProtectionTemplate -Connection $connection -TemplateName TemplateSample1 -
ProtectedObjects $EnterpriseProtectedObject
```

Example: Create a GPO Protection template to protect objects in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

New-CAGPOProtectionTemplate -Connection $connection -TemplateName $templateName -
ProtectedObjects $protectedObjects -OverrideAccounts $overrideAccountDn -
AdminAccounts $adminAccountDn -Credential $forestCredential
```

Get-CAGPOProtectionTemplates

Use this command to see all the GPO protection templates that have been created.

Table 86. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Credential	Credentials used to access the foreign forest.

Example: Get a list of all GPO Protection templates

```
Get-CAGPOProtectionTemplates -Connection $connection
```

Example: Get a list of all GPO Protection templates with objects in a foreign forest

```
$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

Get-CAGPOProtectionTemplates -Connection $connection -Credential $forestCredential
```

Example: Get a list of operations and protected objects for GPO Protection template

```
Get-CAGPOProtectionTemplates -Connection $connection | Where-Object {$_.TemplateName
-eq "TemplateName" } | Select-Object -ExpandProperty ProtectedObjects
```

Set-CAGPOProtectionTemplate

Use this command to modify a GPO protection template.

Table 87. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to be modified.
-ProtectedObjects	List of ProtectedObjects. See New-CAProtectedObject for details.
-Credential (Optional)	Credentials used to access the foreign forest.
-TemplateName (Optional)	The new name for the template.
-DoNotProtectWorkingCopies (Optional)	When enabled, GPOADmin working copies selected for the protection template (or in the AD forest if Enterprise is selected), are ignored by the template. The parameter accepts Boolean $true or $false. **NOTE:** Name-matching is used to identify GPOADmin temporary working copies based on a name prefix "[GPOADmin Working Copy] - ". The service account for the GPOADmin service should also be excluded from protection as an allowed account when the option to exclude GPOADmin working copies is selected.
-OverrideAccounts (Optional)	Accounts allowed or not allowed to change the protected objects.
-OverrideAccountsDenied (Optional)	Specifies if you want to deny the list of user in the OverrideAccounts access. You can specify either $true or $false. Default is false which means that the user accounts are not denied access.
-AdminAccounts (Optional)	Accounts that can manage the protection template. Default is none.
-Disabled (Optional)	Specifies whether the template is enabled or disabled using the Boolean $true or $false.

Example: Modify a GPO Protection template

```
$ProtectedObjects= New-CAProtectedObject -ObjectDistinguishName "distinguishedName"
-Operations Create, Delete, Modify, Link
```

```
$EnterpriseProtectedObject= New-CAProtectedObject -ObjectDistinguishName
"Enterprise" -Operations Create, Delete, Modify, Link

Set-CAADProtectionTemplate -Connection $connection -Template $template -
ProtectedObjects $protectedObject1, $protectedObject2 -AdminAccounts $adminAccountDn
-Schedule $schedule -Disabled $False
```

Example: Modify a GPO template that contains foreign forest objects

```
$ProtectedObjects= New-CAProtectedObject -ObjectDistinguishName "distinguishedName"
-Operations Create, Delete, Modify, Link

$EnterpriseProtectedObject= New-CAProtectedObject -ObjectDistinguishName
"Enterprise" -Operations Create, Delete, Modify, Link

$forestCredential = New-CAForestCredential -ForestName $forestName -Credential
$creds

Set-CAGPOProtectionTemplate -Connection $connection -Template $template -
ProtectedObjects ($ProtectedObjects, $EnterpriseProtectedObject) -
DoNotProtectWorkingCopies $true -OverrideAccounts "distinguishedName" -
OverrideAccountsDenied $true -AdminAccounts "distinguishedName" -Disabled $False -
Credential $forestCredential
```

Remove-CAGPOProtectionTemplate

Use this command to remove a GPO protection template.

Table 88. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The PSCAProtectionTemplate object to remove. Obtain the template objects using the Get-CAGPOProtectionTemplates command and filter to select the object to remove.
-Force	Removes the template without providing confirmation.

Example: Remove a GPO protection template

```
Remove-CAGPOProtectionTemplate -Connection $connection -Template $template
```

About us

Quest creates software solutions that make the benefits of new technology real in an increasingly complex IT landscape. From database and systems management, to Active Directory and Office 365 management, and cyber security resilience, Quest helps customers solve their next IT challenge now. Around the globe, more than 130,000 companies and 95% of the Fortune 500 count on Quest to deliver proactive management and monitoring for the next enterprise initiative, find the next solution for complex Microsoft challenges and stay ahead of the next threat. Quest Software. Where next meets now. For more information, visit www.quest.com.

Our brand, our vision. Together.

Our logo reflects our story: innovation, community and support. An important part of this story begins with the letter Q. It is a perfect circle, representing our commitment to technological precision and strength. The space in the Q itself symbolizes our need to add the missing piece—you—to the community, to the new Quest.

Contacting Quest

For sales or other inquiries, visit www.quest.com/contact.

Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at https://support.quest.com.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request.
- View Knowledge Base articles.
- Sign up for product notifications.
- Download software and technical documentation.
- View how-to-videos.
- Engage in community discussions.
- Chat with support engineers online.
- View services to assist you with your product.