Quest

Quest® Change Auditor 7.0

PowerShell Command Guide

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.
Attn: LEGAL Dept.
4 Polaris Way
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 - March 2019
Software Version - 7.0

Contents

1

PowerShell Commands

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.
-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).

Example: Install a Change Auditor for Active Directory license

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

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 mandatory.

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

```
Find-CASuitableCoordinator –InstallationName 'DEFAULT'
```

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 6. 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.
-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.

Table 7. 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.	`$connection = Connect-CAClient`
Connect to a specific coordinator by computer name and port.	`Connect-CAClient –ComputerName ‘ca-cord.DomainName.com’ –Port 52289`
Connect to the first suitable coordinator in the domain “DomainName.com”.	`Connect-CAClient –DomainName ‘DomainName.com’`
Connect to the first suitable coordinator in the domain “DomainName.com” with an installation name “DEFAULT”.	`Connect-CAClient –DomainName ‘DomainName.com’ –InstallationName ‘DEFAULT’`
Connect to a coordinator found from Find-CACoordinators.	`$coordinators = Find-CACoordinators –DomainName ‘DomainName.com’` `$connection = Connect-CAClient –CoordinatorConnectionPoint $coordinators[0]`

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
```

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 8. 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 9. 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
```

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
```

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 10. 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 11. 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 “01/01/2016 12:00:00”
```

Ping-CAAgent

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

Table 12. 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 13. 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 “01/01/2016 12:00:00”
```

Update-CAAgent

Use this command to upgrade an agent.

Table 14. 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 15. 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 16. 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 17. 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 FluidFS, Azure AD, and Office 365.

Table 18. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to modify.
-Credential (This is only required for the FluidFS module. It is optional for all others.)	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 FluidFS, Azure AD, and Office 365.

Table 19. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to modify.
-Credential (This is only required for the FluidFS module. It is optional for all others.).	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 FluidFS, Azure AD, and Office 365.

Table 20. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to remove.
-Credential (This is only required for the FluidFS module. It is optional for all others.)	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 21. 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 22. 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 Azure Active Directory and Office 365 targets. If your proxy server requires authentication, you must also set the credentials using the -ProxyCredential parameter.

Table 23. 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 24. 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 25. 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 26. 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 27. 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 -Connection $connection
-Configuration $configuration
```

Remove-CATemplatesFromConfiguration

Use this command to remove templates from a configuration.

Table 28. 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.

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 29. 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

```
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 30. 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 31. 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 name, default folder, or limit of a public or private search from the installation.

Table 32. 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 (Optional)	An optional parameter specifying a new name for the search.
-DefaultFolderPath (Optional)	An optional parameter specifying a new default folder path for the search.
-Limit (Optional)	An optional parameter specifying a new limit for the search.
-PassThru (Optional)	A switch that specifies to return the updated search after the command runs.

Example: Changing the display name of a search form “All Owner Mailbox Events” to “Display my owner mailbox events”

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

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

Set-CASearchProperties $connection –Name “Display my owner mailbox events” -PassThru
```

Copy-CASearch

Use this command to copy a search in the installation.

Table 33. 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.
-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 34. 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 35. 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\Skype\" 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\Skype"
```

Remove-CASearch

Use this command to remove a public or private search from 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.
-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 37. 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 38. 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 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 native 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 39. 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 40. 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 41. 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 42. 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 remove.
-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 43. 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 44. 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 Fluid File System auditing

Change Auditor for Fluid File System tracks, audits, reports, and alerts on file and folder changes in real time, translating events into simple text and eliminating the time and complexity required by native auditing. The auditing scope can be set on an individual file or folder or an entire file system recursive or non-recursive. Change Auditor also allows you to include or exclude certain files or folders from the audit scope to ensure a fast and efficient audit process.

Change Auditor captures events and provides detailed information relating to the following activities:

- File and folder access
- File and folder creation, deletion, and renames
- File and folder permission changes
- Content changes, such as file opens and writes

NOTE: Change Auditor for Fluid File System audits only SMB operations on FluidFS clusters.

The following commands are available to manage Fluid File System auditing:

- Get-CAFluidFSClusters
- Get-CAFluidFSEncryptionStatus
- Get-CAFluidFSEventClassInfo
- Get-CAFluidFSTemplates
- Get-CAFluidFSVolumes
- New-CAFluidFSAuditVolume
- New-CAFluidFSTemplate
- Clear-CAFluidFSTemplate
- Set-CAFluidFSTemplate
- Set-CAFluidFSEncryptionCredential

Get-CAFluidFSClusters

Use this command to see a list of all Fluid File Service clusters available to audit.

Table 45. Parameter description

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

Example: See a list of FluidFS cluster names

```
Get-CAFluidFSClusters -Connection $connection
```

Get-CAFluidFSEncryptionStatus

Use this command to see if encryption has been set. Encryption protects the data as it passes between the FluidFS cluster and the Change Auditor agents.

Table 46. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ClusterName	The name of the FluidFS cluster to audit.
-ClusterConfigurationCredential	Administrator credentials to access Enterprise Manager.

Example: Determine if FluidFS encryption has been set (True if the encryption status is set; false otherwise)

```
Get-CAFluidFSEncryptionStatus -Connection $connection -ClusterName $clustername -
ClusterConfigurationCredential $credential
```

Get-CAFluidFSEventClassInfo

Use this command to get a list of all available FluidFS event classes.

Table 47. Parameter description

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

Example: Get a list of all FluidFS event classes.

```
Get-CAFluidFSEventClassInfo -Connection $connection
```

Get-CAFluidFSTemplates

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

Table 48. Parameter description

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

Example: Get a list of all FluidFS templates

```
Get-CAFluidFSTemplates -Connection $connection
```

Get-CAFluidFSVolumes

Use this command to get a list of all volumes on a specified cluster.

Table 49. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ClusterName	The name of the cluster from which to retrieve volume names.
-ClusterConfigurationCredential	Administrator credentials to access Enterprise Manager. This allows the Coordinator to connect with the Enterprise Manager Data Collector service and populate the list of available volumes to audit.

Example: See a list of all available volumes on a cluster

```
Get-CAFluidFSVolumes -Connection $connection -ClusterName $clustername -
ClusterConfigurationCredential $credential
```

New-CAFluidFSAuditVolume

Use this command to define which volumes to audit.

- Inclusions allows you to specify what in the selected volume to be audit.
- Exclusions allow you to refine the settings defined on the Inclusions tab. That is, you can optionally specify the names and paths of any subfolders and files in the selected volume to exclude from auditing.

NOTE: When specifying exclusions with PowerShell, you must specify the volume. This is not required for inclusion masks.

For example:

```
$includePaths = "?folder\**"

$excludeFilePaths = "\vol1\*.tmp"
```

Table 50. Parameter description

Parameter	Description
-Volume	The name of the volume to audit.
-IncludePaths	The folders\files to include. **NOTE:** You can also enter the name of an individual subfolder or file to include. However, if you enter the name of a subfolder, you will only receive events for operations performed against the specified subfolder. You will not receive events for operations performed against any child objects under the specified subfolder.
-EventClasses	The events to audit.
-ExcludeFilePaths (Optional)	The name and path of files to exclude from auditing.
-ExcludeFolderPaths (Optional)	The name and path of subfolders to exclude from auditing.
-Disabled (Optional)	Specifies whether auditing is enabled or disabled on the volume. The disable feature allows you to temporarily stop auditing the specified volume without having to remove the auditing template or individual volume from a template.

Example: Define the volumes to audit

```
$auditVolume = New-CAFluidFSAuditVolume -Volume $volumes[0] -IncludePaths
$includePaths -EventClasses $fluidFSEventClasses -ExcludeFilePaths $excludeFilePaths
-ExcludeFolderPaths $excludeFolderPaths -Disabled $False
```

New-CAFluidFSTemplate

To enable FluidFS auditing in Change Auditor, you must first create an auditing template for each file server to audit. Each auditing template defines the location of the file server, the auditing scope, and the Change Auditor agents that are to receive the events.

NOTE: There can be only 1 FluidFS auditing template per file server. If you want to audit multiple audit paths, use the same template to specify all the paths to audit on the selected file server.

Use this command to create a Fluid File System auditing template.

Returns: A FluidFS template object.

Table 51. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ClusterName	The name of the cluster to audit.
-Agents	The Change Auditor agents that are to receive the FluidFS events. **NOTE:** Specifying multiple agents may provide better performance because the file server will load balance audit events and send each assigned agent events round-robin style. However, the downside is that the 'where' field for FluidFS events may contain any one of these agents. Also, if FluidFS event logging is enabled, events are written on multiple agent servers.
-AuditItems	The volumes and their list of exclusions and inclusions.
-ClusterConfigurationCredential	Administrator credentials to access Enterprise Manager. This allows the Coordinator to connect with the Enterprise Manager Data Collector service and populate the list of available volumes to audit.
-Disabled (Optional)	Specifies whether the template is enabled or disabled.

Example: Create a FluidFS template

```
New-CAFluidFSTemplate -Connection $connection -ClusterName $clustername -Agents
$agents -AuditItems $auditItems -ClusterConfigurationCredential $credential
-Disabled $False
```

Clear-CAFluidFSTemplate

Use this command to delete a FluidFS to delete a template when a connection cannot be made with the FluidFS cluster.

NOTE: Auditing settings must then be removed from the cluster using Enterprise Manager.

Table 52. Parameter description

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to delete.

Example: Delete a template when the Fluid FS cluster cannot be reached

```
Clear-FluidFSTemplate -Connection $connection -Template $template
```

Set-CAFluidFSTemplate

Use this command to edit an existing Fluid File System template.

NOTE: You can also use the Enable-CAAgentTemplate and Disable-CAAgentTemplate to enable or disable the template.

Table 53. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-Template	The template to edit.
-ClusterConfigurationCredential	Administrator credentials to access Enterprise Manager. This allows the Coordinator to connect with the Enterprise Manager Data Collector service and populate the list of available volumes to audit.
-Agents (Optional)	The Change Auditor agents that are to receive the FluidFS events.
-AuditItems (Optional)	The volumes and their list of exclusions and inclusions.
-Enable (Optional)	Set to true or false to enable and disable the template.

Example: Modify a FluidFS template

```
Set-CAFluidFSTemplate -Connection $connection -Template $template -
ClusterConfigurationCredential $credential -Enable True -Agents $agents -AuditItems
$auditItems
```

Set-CAFluidFSEncryptionCredential

Use this command to enable or disable encryption for auditing on the Fluid File System cluster. Encryption allows you to protect the event traffic as it passes between the FluidFS cluster and the Change Auditor agents.

Table 54. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-ClusterName	The name of the cluster to audit.
-ClusterConfigurationCredential	Administrator credentials to access Enterprise Manager.
-EncryptionCredential	The service account credentials for the cluster to use when encrypting events.

Example: Modify the encryption (required or not) for auditing on the Fluid FS cluster

```
Set-CAFluidFSEncryptionCredential -Connection $connection -ClusterName $clustername -
ClusterConfigurationCredential $credential -EncryptionCredential $EncryptionCredential
```

Managing Azure Active Directory auditing

Change Auditor audits activity in the Azure portal that corresponds to the events in the Azure Active Directory auditing logs and sign-in activity. Managing Azure Active Directory 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 Azure Active Directory remains. You can delete the web application using the Azure management portal. 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 Azure Active Directory and Office 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 Azure Active Directory.

Table 55. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent is used for Azure Active Directory auditing. **NOTE:** The agent must be allowed to connect to Azure Active Directory. • 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 Azure Active Directory 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 Azure Active Directory auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-WebAppCreationCredential	Azure Active Directory credentials required to create an Azure web application. The credential object is obtained by using the Get-Credential command. **NOTE:** The account must be a user with the Global Administrator role. **NOTE:** When using this parameter, you cannot also specify the –WebAppCreationCredential parameter. **NOTE:** If your tenant does not include an Office 365 subscription, use the -WebAppId and -WebAppKey options described in Create a template using an existing web application.
-AuditLogs	Specifies whether or not to audit the Azure Active Directory 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 Azure Active Directory sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns 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 Azure Active Directory.

Example: Creating Azure Active Directory auditing template using Global Administrator credentials that will collect events generated 30 days in the past.

```
New-CAAzureADTemplate -Connection $connection -WebAppCreationCredential $azureCreds
-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 Azure web application that Change Auditor will use for authentication.

For details on integrating applications with Azure Active Directory and creating a web application, consult the Microsoft documentation. When creating a web application in the Azure Classic Portal, 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 Azure web application:

Table 56. Required permission

System	Permissions
Windows Azure Active Directory	Application Permissions: • Read directory data Delegated Permissions: • Read directory data • Sign in and read user profile

Table 57. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent will be used for Azure Active Directory auditing. **NOTE:** The agent must be allowed to connect to Azure Active Directory. • 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 Azure Active Directory 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 Azure Active Directory auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-Tenant	The Azure Active Directory tenant/directory that you want to audit (for example: yourTenantName.onmicrosoft.com).
-AuditLogs	Specifies whether or not to audit the Azure Active Directory 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 Azure Active Directory sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-WebAppId	An Azure Active Directory web application Id. This application is needed for Change Auditor to authenticate to your Azure Active Directory tenant. **NOTE:** Azure Active Directory and Office 365 must each have their own dedicated web application. **NOTE:** When using this parameter, you cannot also specify –WebAppCreationCredential 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 Azure Active Directory 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 Azure Active Directory.

Example: Creating an Azure Active Directory 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 Azure Active Directory 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 58. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained using the Get-CAAgents command. The agent will be used for Azure Active Directory auditing. **NOTE:** The agent must be allowed to connect to Azure Active Directory. • 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 Azure Active Directory 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 Azure Active Directory 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.
-AuditLogs	Specifies whether or not to audit the Azure Active Directory 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 Azure Active Directory sign-in activity. You must enable at least one type of activity to audit using the - AuditLogs or -SignIns parameter.
-WebAppId	An Azure Active Directory web application Id. This application is needed for Change Auditor to authenticate to your Azure Active Directory tenant. **NOTE:** Azure Active Directory and Office 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 Azure Active Directory tenant.

Example: Modify Azure Active Directory 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 Azure Active Directory templates available within your installation.

Table 59. Available parameters

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

Example: Get a list of all Azure AD 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
- Get-CAO365Templates
- Set-CAO365Template
- Get-CAO365ExchangeMailboxes
- Add-CAO365ExchangeTemplateMailboxes
- Remove-CAO365ExchangeTemplateMailboxes
- Get-CAO365ExchangeTemplateMailboxes

NOTE: When you delete a template (see Remove-CAAgentTemplate), the web application created in Azure Active Directory remains. You can delete the web application using the Azure management portal. 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 Azure Active Directory and Office 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 60. Available parameters

Parameter	Description
-AgentInfo	An agent obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Azure. • 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 Azure Active Directory 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 Azure Active Directory auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-WebAppCreationCredential	Azure Active Directory account credentials required to create an Azure web application. The credential object is obtained by using the Get-Credential command. **NOTE:** The account must be a user with the Global Administrator role.
-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.
-Disabled (Optional)	Specifies whether the auditing template 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.

Parameter	Description
-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.
-O365ExchangeAdminCredential (Optional)	An account with the Exchange Administrator role. It is used to configure the mailbox auditing settings in the tenant that are defined in the template (such as enabling auditing of owner activity). These credentials are used periodically by the agent to validate or update auditing settings, so they are securely stored in Change Auditor. **NOTE:** The account must belong to the same tenant as the Azure Active Directory account used to create the Azure web application.

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-CAO36Template -Connection $connection -WebAppCreationCredential $azureCreds -
AgentInfo $agent -O365ExchangeAdminCredential $o365Creds -AuditAdministration $true
–AuditOrganization $true –HistoricalEventCollectionDays 7
```

Create a template using an existing web application

Alternatively, use these parameters when using a pre-created Azure web application that will be used by Change Auditor for authentication.

For details on integrating applications with Azure Active Directory and creating a web application, consult the Microsoft documentation. When creating a web application in the Azure Classic Portal, 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 Azure web application:

Table 61. Required permission

System	Permissions
Windows Azure Active Directory	Application Permissions: • Read directory data Delegated Permissions: • Read directory data • Sign in and read user profile
Office 365 Management APIs	Application Permissions: • Read activity reports for your organization • Read activity data for your organization • Read service health information for your organization Delegated Permissions: • Read activity reports for your organization • Read activity data for your organization • Read service health information for your organization

Table 62. Available parameters

Parameter	Description
-AgentInfo	An agent object obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Azure. • 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 Azure Active Directory 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 Azure Active Directory auditing. This is the port that is used for communicating with the tenant.
-Connection	A connection obtained by using the Connect-CAClient command.
-Tenant	The Azure AD tenant/Directory that you would like Change Auditor to audit (for example: yourTenantName.onmicrosoft.com).
-WebAppId	An Azure Active Directory web application Id. This application is needed for Change Auditor to authenticate to your Azure Active Directory tenant. **NOTE:** Azure Active Directory and Office 365 must each have their own dedicated web application. **NOTE:** When using this parameter, you cannot also specify the –WebAppCreationCredential 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 Azure Active Directory tenant. **NOTE:** When using this parameter, you cannot also specify the –WebAppCreationCredential parameter.
-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.
-Disabled (Optional)	Specifies whether the auditing template 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.
-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.

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 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.
-O365ExchangeAdminCredential (Optional)	An account with the Exchange Administrator role. It is used to configure the mailbox auditing settings in the tenant that are defined in the template (such as enabling auditing of owner activity). These credentials are used periodically by the agent to validate or update auditing settings, so they are securely stored in Change Auditor. **NOTE:** The account must belong to the same tenant as the Azure AD account used to create the Azure web application.

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 -AgentInfo $agent -
O365ExchangeAdminCredential $o365Creds -WebAppKey $webAppKey -WebAppId $webAppId -
Tenant $tenant -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.
-WebAppCreationCredential	Azure Active Directory account credentials required to create an Azure web application. The credential object is obtained by using the Get-Credential command. **NOTE:** The account must be a user with the Global Administrator role. **NOTE:** When you specify this parameter a new web application is created and assigned to the template.
-WebAppId	An Azure Active Directory web application Id. This application is needed for Change Auditor to authenticate to your Azure Active Directory tenant. **NOTE:** Azure Active Directory and Office 365 must each have their own dedicated web application. **NOTE:** When using this parameter, you cannot also specify the –WebAppCreationCredential 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 Azure Active Directory tenant. **NOTE:** When using this parameter, you cannot also specify the –WebAppCreationCredential parameter.

Parameter	Description
-AgentInfo (Optional)	An agent object obtained by using the Get-CAAgents command. **NOTE:** The agent must be able to connect to Azure. • 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 Azure Active Directory 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 Azure Active Directory 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.
-EnableExchange (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.
-O365ExchangeAdminCredential (Optional)	An account with the Exchange Administrator role. It is used to configure the mailbox auditing settings in the tenant that are defined in the template (such as enabling auditing of owner activity). These credentials are used periodically by the agent to validate or update auditing settings, so they are securely stored in Change Auditor. **NOTE:** The account must belong to the same tenant as the Azure AD account used to create the Azure web application.

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: Replace the web application

```
Set-CAO365Template  -Connection $connection -Template $template -WebAppId $webAppId
-WebAppKey $webAppKey
```

Example: Replace the agent

NOTE: Replacing the agent requires additional parameters to provide authentication credentials and tenant information.

```
Set-CAO365Template  -Connection $connection -Template $template -AgentInfo $agent -
WebAppId $webAppId -WebAppKey $webAppKey -O365ExchangeAdminCredential $o365LiveCreds
```

Get-CAO365Templates

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

Table 63. 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
```

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 64. 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 65. 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-CAO365TemplateMailboxes -Connection $connection -Template $template -Mailboxes
$mailboxes -AuditOwnerEvents
```

Remove-CAO365ExchangeTemplateMailboxes

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

Table 66. 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 67. 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 “John S”

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

Managing Skype for Business auditing

The following commands are available to manage Skype for Business auditing:

- Get-CASkypeEventClassInfo
- New-CASkypeTemplate
- Get-CASkypeTemplates
- Set-CASkypeTemplate
- Remove-CASkypeTemplate

Get-CASkypeEventClassInfo

Use this command to see the list of event classes available for the Skype for Business subsystem.

Table 68. Available parameters

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

Example: Get a list of all available Skype for Business event classes

```
Get-CASkypeEventClassInfo -Connection $connection
```

New-CASkypeTemplate

Use this command to add a Skype for Business template to Change Auditor.

NOTE: It is recommended that the Change Auditor Coordinator Service is running in the same forest as the Skype for Business Central Management Store (CMS) database server.

Once the template has been created, the agent is notified of the Skype for Business Central Management Store details and the events to audit.

Table 69. Available parameters

Parameter	Description
-AgentInfo	The Change Auditor agent to audit the Skype events. This agent must be executing on the Skype for Business Central Management Store database server.
-AuditItems	Collection of events to audit.
-Connection	A connection obtained by using the Connect-CAClient command.
-CMSInstanceName (Optional)	The Microsoft Skype for Business server 2015 / Microsoft Lync Server 2013 Central Management Store (CMS) SQL Server Instance Name. The CMS Instance name must be provided only when the Change Auditor Coordinator Service is not in the same Active Directory forest as Microsoft Skype for Business Server 2015 / Microsoft Lync Sever 2013.
-DatabaseCMSCredential	Skype for Business Central Management Store database credentials.
-TemplateName	The name of the template.
-UseWindowsAuthentication	Specifies whether to use Windows authentication when connecting to the Central Management Store database. If Windows authentication is not used, SQL Authentication will be used.
-SkipCMSDatabaseConnectivityTest (Optional)	Specifies whether to test the Central Management Store (CMS) SQL Server Connection using the supplied CMS credentials.
-Disabled (Optional)	Specifies whether the template is disabled.

Example: Create a Skype for Business template

```
New-CASkypeTemplate -AgentInfo $agentInfo -AuditItems $auditItems -Connection
$connection -DatabaseCMSCredential $dbCredential -TemplateName 'Skype for Business
Template' -UseWindowsAuthentication $True -Disabled $False
```

Get-CASkypeTemplates

Use this command to see all the Skype for Business templates that have been created.

Table 70. Available parameters

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

Example: Get a list of all Skype for Business templates

```
Get-CASkypeTemplates -Connection $connection
```

Set-CASkypeTemplate

Use this command to update the properties of an existing Skype for Business template. Once the template has been updated, the agent is notified of the Skype for Business Central Management Store details, and the events to audit.

NOTE: If the Central Management Store database server has changed, you need to redeploy your agent to the new Central Management Store SQL Server, delete the existing template, then create a new template.

Table 71. Available parameters

Parameter	Description
-AgentInfo	The Change Auditor agent to audit the Skype events. This agent must be executing on the Skype for Business Central Management Store database server.
-AuditItems	Collection of events to audit.
-Connection	A connection obtained by using the Connect-CAClient command.
-DatabaseCMSCredential	Skype for Business Central Management Store database credentials.
-Template	The name of the existing template to update.
-TemplateName	The name of the template.
-UseWindowsAuthentication	Specifies whether to use Windows authentication when connecting to the Central Management Store database. If Windows authentication is not used, SQL Authentication will be used.
-SkipCMSDatabaseConnectivityTest (Optional)	Specifies whether to test the Central Management Store (CMS) SQL Server Connection using the supplied CMS credentials.
-Disabled (Optional)	Specifies whether the template is disabled.

Example: Modify a Skype for Business template

```
Set-CASkypeTemplate -Connection $connection -Template $templateToUpdate 'Updated Skype for Business Template' -AgentInfo &agentInfo -AuditItems &$auditItems -DatabaseCMSCredential $dbCredential -UseWindowsAuthentication $True -Disabled $False
```

Remove-CASkypeTemplate

Use this command to remove a Skype for Business template. Agents associated with the template would be notified and Skype for Business configuration events would not be audited anymore.

Table 72. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-TemplateName	The name of the template to remove.

Example: Remove a Skype for Business template

```
Remove-CASkypeTemplate -Connection $connection -TemplateName 'Skype For Business Template'
```

Working with protection templates

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

The following commands are available to manage Active Directory protection:

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

New-CAADProtectionTemplate

Use this command to create an Active Directory protection template.

Table 73. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command.
-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 -Connection $connection -
ObjectDistinguishName “ObjectName” -ProtectedScope ScopeObject -Operations Create

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

New-CAProtectedObject

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

Table 74. Available parameters

Parameter	Description
-ObjectDistinguishName	Distinguish name of object to protect.
-ProtectedScope	Scope of coverage for the protected object. Specify the scope using one of the following values: • ScopeObject • ScopeOneLevel • ScopeSubtree
-Operations	Operations to be denied for the selected object: • None • Create • Modify • Delete • Move **NOTE:** You can specify multiple operations.

Example: Create a new protected object

```
New-CAProtectedObject -Connection $connection -ObjectDistinguishName "ObjectName" -
ProtectedScope ScopeObject -Operations Create
```

New-CAScheduledTimeRange

Use this command to schedule when to enforce the protection.

Table 75. 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.

Table 76. Available parameters

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

Example: Get a list of all Active Directory Protection templates

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

Remove-CAADProtectionTemplate

Use this command to remove an Active Directory protection template.

Table 77. 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-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
```

About us

Quest provides software solutions for the rapidly-changing world of enterprise IT. We help simplify the challenges caused by data explosion, cloud expansion, hybrid datacenters, security threats, and regulatory requirements. We are a global provider to 130,000 companies across 100 countries, including 95% of the Fortune 500 and 90% of the Global 1000. Since 1987, we have built a portfolio of solutions that now includes database management, data protection, identity and access management, Microsoft platform management, and unified endpoint management. With Quest, organizations spend less time on IT administration and more time on business innovation. 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.