One Identity Data Governance Edition 9.2

Technical Insight Guide

The information in this document is provided in connection with One Identity products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of One Identity LLC products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, ONE IDENTITY ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ONE IDENTITY BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ONE IDENTITY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. One Identity makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. One Identity does not make any commitment to update the information contained in this document.

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

One Identity LLC.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

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

Patents

One Identity is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at http://www.OneIdentity.com/legal/patents.aspx.

Trademarks

One Identity and the One Identity logo are trademarks and registered trademarks of One Identity LLC. in the U.S.A. and other countries. For a complete list of One Identity trademarks, please visit our website at www.OneIdentity.com/legal. All other trademarks are the property of their respective owners.

Legend

> ⊗ **WARNING: A WARNING icon highlights a potential risk of bodily injury or property damage, for which industry-standard safety precautions are advised. This icon is often associated with electrical hazards related to hardware.**

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

Data Governance Edition Technical Insight Guide
Updated - 15 September 2023, 10:42
Version - 9.2

Contents

1

Introduction

The One Identity Manager Data Governance Edition Technical Insight Guide is intended for IT specialists who are involved in the Data Governance Edition deployment, configuration and maintenance. It provides a technical insight into the product components, operations and processes. It is written for advanced audiences who want a deeper understanding of the Data Governance Edition components and how they communicate with each other.

This document is intended to cover the basic functionality and technology of Data Governance Edition. It is not intended as a stand-alone document and makes references to supporting documentation that should be used when deploying the product in your production environment.

Available documentation

Data Governance Edition documentation includes the following manuals:

- *One Identity Manager Data Governance Edition User Guide*

  This guide includes Data Governance Edition administration information.

- *One Identity Manager Data Governance Edition Deployment Guide*

  This guide includes Data Governance Edition installation, configuration, and deployment information.

- *One Identity Manager Data Governance Edition IT Shop Resource Access Requests User Guide*

  This guide includes details about the self-service resource requests related to resources that are governed, including the file system share creation request in the IT Shop.

- *One Identity Manager Data Governance Edition Technical Insight Guide*

  This guide is intended for advanced audiences who want a deeper understanding of the Data Governance Edition components and how they communicate with each other. It also provides a description of the configuration file settings, registry key settings and PowerShell commands.

Online versions of the Data Governance Edition guides are available on the technical support web portal: https://support.oneidentity.com/identity-manager-data-governance-edition/technical-documents

For supporting One Identity Manager information, see the One Identity Manager documentation. Online versions of the One Identity Manager guides are available on the technical support web portal: https://support.oneidentity.com/identity-manager/technical-documents

Data Governance Edition network communications

Data Governance Edition uses a Service Connection Point (SCP) to locate the Data Governance service, listening ports for communication between Data Governance components, and network encryption to secure sensitive information.

Service Connection Points

Data Governance Edition publishes a Service Connection Point (SCP) in Active Directory so the Data Governance configuration wizard and Data Governance agents can locate the Data Governance service. Upon startup, the Data Governance service searches Active Directory in order to verify that the SCP is correct. When the Data Governance configuration wizard or agents start up, they search Active Directory for the SCP objects within their Active Directory forest, in order to retrieve connection information from the Data Governance service such as host name, listening port, deployment name, and other authentication information. The agents use the deployment name in the keywords search so they will only find services with the same user configured DGE deployment name.

The SCP objects are published directly subordinate to the service's computer object in Active Directory. SCP objects can be viewed and updated using Microsoft's ADSI Edit MMC snap-in.

The Data Governance service installs and maintains a single SCP (CN=DataGovernance.Server). The service checks and updates the Active Directory objects each time the service starts up.

CN=DataGovernance.Server

The Data Governance service SCP contains the following key elements, which are stored in its Active Directory attributes.

Table 1: DataGovernance.Server SCP

Attribute/Attribute Syntax	Function	Default Value
CN Attribute syntax: String	SCP Name	DataGovernance.Server
keywords Attribute syntax: Multi-valued string	Used to store the following information to facilitate locating the SCP: • Database: Resource Activity database name (for example, DGE_DEFAULT) • DeploymentName • serverDNSName • serviceClassName • siteName • version	
serviceBindingInformation Attribute syntax: Multi-valued string	Contains the default tcp.net port and HTTP port	<XML>
serviceClassName Attribute syntax: String	Used to store service class for authentication	DataGovernance.Server
serviceDNSName Attribute syntax: String	FQDN of the computer running the Data Governance service	<Server FQDN>
serviceDNSNameType Attribute syntax: String	The DNS record type of the host listed in the serviceDNSName	A

Data Governance Edition required ports

> NOTE: For agent deployments, open the following file and printer sharing ports:
>
> - TCP 135
> - UDP 137
> - UDP 138
> - TCP 139
> - TCP 445

Table 2: Ports required for communication

Port	Direction	Description
8721	Incoming	TCP (HTTP) port opened on the Data Governance server computer. This is the base port for the Data Governance REST API, used for communication with Data Governance server REST services, including the One Identity Manager clients and Windows PowerShell.
8722	Incoming	TCP (net.tcp) port opened on the Data Governance server computer. Used for communication with Data Governance agents, One Identity Manager clients, One Identity Manager web server, and PowerShell. NOTE: The net.tcp port is configurable in the Data Governance Configuration wizard. The HTTP port (8721) listed above should always be 1 less than the net.tcp port. These first two ports align with the base addresses in the DataGovernanceEdition.Service.exe.config file under the IndexServerHost service. It is highly recommended that you only change this port using the Data Governance Configuration wizard to ensure the configuration file, One Identity Manager database and service connection points are updated properly; otherwise, you may lose connection with the Manager, the Data Governance service and/or Data Governance agents. IMPORTANT: Do NOT use the Designer to change the QAMServer configuration parameters, including the Port parameter.
8723	Incoming	HTTP port used for communication with the One Identity Manager web server (/landing and /home pages).
18530 - 18630	Incoming	TCP port range opened on all agent computers. Used for communication with the Data Governance server. (The first agent on an agent host will use port 18530, and each subsequent agent on the same host will take the next available port, i.e., 18531, 18532, and so on.). In addition, this range is used to open a TCP

Port	Direction	Description
		listener for NetApp Cluster Mode hosts if resource activity collection is enabled.

Component communication

Server and database communication

Information about all Data Governance Edition infrastructural elements such as service accounts, managed hosts and the security index information collected by the Data Governance agents is stored in the One Identity Manager database. Processing of security index updates, access and activity queries or any infrastructural changes to the system involve communication between the Data Governance server and the database.

How is the database connection information stored securely?

The connection information used when communicating with the One Identity Manager database is stored in the Windows Registry on the Data Governance server. The connection information is written to the registry key "HKLM\SOFTWARE\One Identity\Broadway\Server" and is encrypted using the Microsoft Data Protection API.

Only the user account that encrypts the value can read it. If the account running the Data Governance server is changed, the database connection string has to be reset and re-encrypted.

Agent and server communication

Data Governance agents are semi-autonomous services running in a distributed environment. They are designed to remain fault tolerant in a fluctuating global network. In a typical organization, computers are rebooted, network outages occur, and systems are disrupted in any number of ways. Data Governance agents are set to automatically start when a server is restarted. Data Governance agents require an initial configuration from the server; however, they will continue to scan and collect activity per configuration even when unable to communicate with the Data Governance server. All the collected activity and security updates are synchronized with the Data Governance server when connectivity is restored.

How is this communication encrypted?

The communication uses encrypted WCF (Windows Communication Foundation) channels and the net.tcp protocol. .NET v4.5 is required on all agent host computers, except for SharePoint 2010 agents, which requires .NET v3.5.1.

Client and server communication

Data Governance client elements are embedded into the Manager client application. The user interface elements communicate with the Data Governance server and directly with the One Identity Manager database as needed.

Communication with the database is performed in the same way as any other One Identity Manager database communication, using the authentication information provided when the user launches the client tools.

When communicating with the Data Governance server, the client uses an encrypted WCF channel and the net.tcp protocol.

.NET 4.5.2 is required on the Data Governance server and client computers.

How is this communication authenticated?

When communicating directly with the One Identity Manager database, the client is authenticated using standard One Identity Manager authorization checks. For more information on this type of authentication, see Granting Access Permissions to One Identity Manager Schema in the *One Identity Manager Configuration Guide*.

When user interface elements communicate with the Data Governance server, the authentication is performed using the One Identity Manager role-based authentication checks using the logged on Windows identity. This can lead to a discrepancy in authentication between the client and server. If possible, it is recommended that the client user authenticates to One Identity Manager using the “Active Directory user account (role based)” authentication mechanism, so no ambiguity exists. This mechanism maps the logged on Active Directory account to a One Identity Manager identity and uses that identity’s application roles to determine what permissions they have.

> NOTE: Regardless of the identity used to log in to the client application, it is the Identity associated with the logged in Windows account that is used for permissions checks when communicating with the Data Governance server.

Communication segments

This table describes each segment of communication that occurs in the Data Governance Edition system along with technical details for each type of communication.

Table 3: Data Governance Edition communication segments

From / To	Originating port	Protocol	Destination port
Data Governance service to One Identity Manager database Actions involved:	Dynamic	TCP	SQL Server port NOTE: A request may go through the One Identity

From / To	Originating port	Protocol	Destination port
- Any queries or data manipulation that may be required. - Inserting of new data and selecting data to display in the Manager client.			Manager Application Server if configured, instead of directly to the database.
Data Governance service to Resource Activity database Actions involved: - Any queries or data manipulation that may be required. - Inserting of new data and generating reports on existing data.	Dynamic	TCP	SQL Server NOTE: A request may go through the One Identity Manager Application Server if configured, instead of directly to the database.
One Identity Manager service (job server) to Data Governance service Actions involved: - Web service requests for self-service access.	Dynamic	TCP	Specified by customer during installation. Default value is 8722.
Data Governance service to Windows Server on which to install agent Actions involved: - Deploy agent. - Uses the associated domain service account to copy installation files to a destination Windows Server using that server's administrative share (Admin$).	Dynamic	SMB	445
Data Governance service to agent service Actions involved: - Notify agent of an	Dynamic	TCP (using Windows authentication of the "Log On As" account of	Next unused port from the configured "BaseActivePort". Default value of "BasesActivePort" is

From / To	Originating port	Protocol	Destination port
awaiting command. - The only thing the Data Governance service sends an agent service, unsolicited, is command messages. The agent then processes the command message and may initiate a request back to the server to get additional data that is associated with the command.		the Data Governance Service Windows Service)	18530.
Agent to Data Governance service Actions involved: - Connection, Keep-Alive/Status, Queries/Reports. - An agent initiates the connection on startup. It periodically sends keep-alive and status messages as well as synchronization.	Dynamic	TCP (using Windows authentication of the "Log On As" account of the agent's Windows Service)	Specified by customer during installation. Default value is 8722.
Data Governance service to NetApp 7-Mode device with CIFS or NFS file system protocols enabled Actions involved: - Configure FPolicy on NetApp 7-Mode filer. - Upon deployment of a managed host in 7-Mode, the Data Governance service connects to the NetApp filer and creates/configures an FPolicy if real-time security updates or resource activity collection is	Dynamic	RPC (using Windows authentication of the "Log On As" account of the Data Governance Windows Service)	Named pipe on NetApp filer: *<Host Name*>\pipe\NETAPPSVC

From / To	Originating port	Protocol	Destination port
enabled. This does not apply to NetApp Cluster Mode.			
Data Governance service to NetApp 7-Mode or Cluster device with NFS file system protocol enabled Actions involved: - Browse resources. - When configuring the managed paths for a managed host, or using the Resource browser to browse the file system.	Dynamic	HTTPS (using the username and password specified in the managed host configuration)	443
Agent to NetApp 7-Mode device with CIFS or NFS file system protocols enabled Actions involved: - Configure FPolicy on NetApp 7-Mode filer. - Upon startup, establish a connection to the NetApp device if real-time security updates or resource activity collection is enabled.	Dynamic	RPC (using Windows authentication of the "Log On As" account of the agent's Windows Service.	Named pipes on NetApp filer: <*Host Name*>\pipe\NETAPPSVC and <*Host Name*>\pipe\ntapfpcp
NetApp 7-Mode to agent Actions involved: - NetApp sends file screen requests when real-time security updates or resource activity collection is enabled. - The agent listens to a named pipe for incoming screen request messages from the NetApp filer for any monitored file system	Dynamic	RPC	Named pipe: <\pipe\ntapfprg_<*Agent Instance ID*>

From / To	Originating port	Protocol	Destination port
events.			
Agent to NetApp Cluster Mode with CIFS or NFS file system protocols enabled Actions involved: • Configure FPolicy on NetApp Cluster mode filer. • The NetApp Data LIF on which the file share is exposed must be the destination when resolving the host name. Also, the "Management Access" setting must be enabled on the LIF.	Dynamic	HTTPS	443
NetApp Cluster Mode to Agent Actions involved: • NetApp sends file screen requests when real-time security updates or resource activity collection is enabled. • The agent listens on a TCP port for incoming screen request messages from the NetApp filer for any monitored file system events.	Dynamic	TCP	Next unused port from the configured "BaseActivePort". Default value of "BasesActivePort" is 18530.
Agent to NetApp device with CIFS file system protocol enabled Actions involved: • File system scanning. • The agent collects security information on all files and folders in the specified managed paths.	Dynamic	CIFS/SMB (using Windows authentication of the "Log On As" account of the agent's Windows Service)	445

From / To	Originating port	Protocol	Destination port
Data Governance service to EMC Celerra device Actions involved: - View/update cepp.conf. - When real-time security updates or resource activity collection is enabled, you must configure the cepp.conf file on the EMC device.	Dynamic	SSH	22
Data Governance service to EMC Isilon device with NFS file system protocol enabled Actions involved: - Browse resources. - When configuring the managed paths for a managed host, or using the Resource browser to browse the file system.	Dynamic	HTTPS (using the username and password specified in the managed host configuration)	Specified by customer when configuring managed host. Default value is 443.
Agent service to EMC device with CIFS file system protocol enabled Actions involved: - File system scanning. - The agent collects security information on all files and folders in the specified managed paths.	Dynamic	CIFS/SMB (using Windows authentication of the "Log On As" account of the agent's Windows Service)	445
Agent service to EMC Isilon device with NFS file system protocol enabled Actions involved: - File system scanning. - The agent collects security information on all files and folders in the	Dynamic	HTTPS (using the username and password specified in the managed host configuration)	Specified by customer when configuring managed host. Default value is 443.

From / To	Originating port	Protocol	Destination port
specified managed paths.			
Agent to SharePoint SQL Server database Actions involved: - Resource scanning. - Connects directly to the SharePoint SQL Server database on the local machine to perform resource scanning.	Dynamic	TCP	Default SQL Server port, typically 1433.
Data Governance service to Cloud API Actions involved: - Browse resources. - When configuring the managed paths for a managed host, or using the Resource browser to browse for resources.	Dynamic	REST over HTTP with OATH authentication	Dynamic
Agent to Cloud API Actions involved: - Resource scanning. - Upon startup, the agent collects all team groups and their members. Thereafter, this scan is performed once a day by default. The agent synchronizes to the server only if there is a change. - The agent collects security information of all files and folders in the specified managed paths. NOTE: Managed paths are selected within the scope of the administrator on OneDrive	Dynamic	REST over HTTP with OATH authentication	Dynamic

From / To	Originating port	Protocol	Destination port
for Business managed hosts.			
Web client to Data Governance service Actions involved: • Web service requests for self-service access.	Dynamic	TCP	Specified by customer during installation. Default value is 8722.
Windows PowerShell to Data Governance service Actions involved: • Data Governance API • Use the Data Governance API via web service requests to automate tasks or add custom behavior.	Dynamic	TCP	Specified by customer during installation. Default value is 8722.

Data Governance service

The *One Identity Manager Data Governance Edition Deployment Guide* provides detailed steps explaining how to deploy the Data Governance service; the information provided here is intended to provide some additional information for those interested in the internal functions of this process and the Data Governance service.

Data Governance Edition deployment process

The deployment process for the Data Governance service includes the following:

- The Data Governance installer deploys and configures the Data Governance service.
- The Data Governance configuration wizard creates and initializes the Resource Activity database.
- Connection strings to the One Identity Manager database and Resource Activity database are encrypted and stored in the registry on the Data Governance service machine.
- The Data Governance service creates and publishes a Service Connection Point (SCP) in Active Directory so the Data Governance configuration wizard, server and agents can locate the Data Governance service.
- Configuration parameters are set in the One Identity Manager database.
- In the absence of One Identity Manager target system synchronization, the Data Governance service automatically harvests the forest topology, including:
  - Creating Identity records for all members found in each domain's Domain Admin group.

- Creating an Identity record for the current account running the Data Governance configuration wizard.
- Linking these accounts to the correct Data Governance application roles.

It is highly recommended that you use the Data Governance Configuration wizard to install the Data Governance service and Resource Activity database. If however, you need to install the Data Governance service to a different location other than the default location, you can use the Windows installer that is provided. For more information, see Manually deploying Data Governance service on page 31.

Data Governance service configuration

Data Governance service configuration settings are stored in one of the following places:

- The Data Governance service contains settings in the DataGovernanceEdition.Service.exe.config file in the server directory: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server.

  For more information on the Data Governance service configuration file settings that can be configured, see Configurable configuration file settings.
- Some Data Governance service settings can also be set in the Windows registry, under the following keys:
  - HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
  - HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client

  For more information on the Data Governance service registry settings that can be configured, see Data Governance service registry settings.

Data Governance service internal tasks

The following table lists the internal tasks that the Data Governance service performs, including the internal service name, a brief description of the task and the configuration variables that are available to customize the task.

Table 4: Data Governance service internal tasks

Internal service name	Task description
AccessQueriesService	Handles all resource access queries. This includes retrieving all trustees with access to a given resource, as well as all resources a given trustee has access to.
AccessSelfService	Handles the self-service requests initiated from the IT

Internal service name	Task description
	Shop. This includes identifying best fit groups based on resource and access requirements, retrieving group information, and getting or setting self-service configuration options. **Configuration settings**: - SelfService.MaximumMethodsCount: Maximum number of self-service groups that can be returned for consideration. Default: 5 - SelfService.SuitabilityThreshold: The lower bounds for suitability used in returning self-service groups. Default: 100
AgentLeaseManager	Handles all aspects of agent lease management. This includes registering and unregistering agents, renewing leases, verifying agent connectivity, and retrieving agent information. The service manages lease renewal over a given period of time (configurable in the application configuration) by checking for expired agent leases and setting the agent states accordingly. The Data Governance service uses this internal service to determine what agents are functioning. If the server does not receive a lease renewal from an agent in the expected time frame, the agent goes into the "Lease Expired" state. This indicates that the server is unable to receive information from the agent. **Configuration setting**: - AgentLeaseRenewPeriod: The refresh rate for checking for expired leases. Default: 5 minutes
DFSDataSync	Synchronizes managed DFS host information into the One Identity Manager database. This process enumerates the DFS targets and stores the relevant information within the database. Synchronization is performed using the service account linked to the managed host being synchronized. The information is harvested on a regular bases, based on the configuration variable. **Configuration setting**: - DfsDataSyncInterval: The interval that defines when a DFS synchronization occurs. Default: 1440 minutes (1 day)
EnterpriseBrowsing	Is used for getting and setting resource security,

Internal service name	Task description
	retrieving domain credentials, service account retrieval, SID and trustee resolution, and resource enumeration.
GroupResolutionService	Is used for a number of services, including group expansion, domain retrieval, group searches, data model retrieval, and SID retrieval. In addition, this service maintains a cache of known managed domains and security information that is refreshed regularly based on configuration variables. For group expansion, the service account for the managed domain is used; however if this fails, the account used for Active Directory synchronization is used instead. In this case, the account used for Active Directory synchronization should be granted log on as service rights to the Data Governance server. **Configuration settings**: • SyncDomainPasswordInterval: The interval that defines when the managed domain and security information cache is refreshed. Default: 60 seconds
IndexServer	Provides the framework for processing messages received from deployed agents. This is purely internal framework and there are no configuration parameters.
InfrastructureManagement	Is used for general infrastructure management. This includes actions such as triggering collection of data under governance and handling the steps required when a service is updated. The service also contacts the agent to retrieve points of interest (POI) information on governed resources on a regular interval based on configuration variables. **Configuration settings**: • CollectPoi.MaxConcurrentQueries: The number of simultaneous queries that can be performed. Default: 5 queries • CollectPoi.CheckFrequencyInMinutes: The frequency at which Data Governance Edition checks for slate POI information. Default: 10 minutes • CollectPoi.OverdueThresholdInMinutes: The

Internal service name	Task description
	amount of time before a resource is considered to be overdue for POI collection, and a collection is performed. Default: 1440 minutes (1 day) • CollectPoi.QueryTimeoutInMinutes: The amount of time before a POI query expires. Default: 360 minutes • CollectPoi.QueryUpperBound: The result limit for a POI query. Default: 1,000,000 resources • CollectPoi.QueryBatchSize: The threshold at which a query is sent to the agent. The number of identified POIs must be greater than or equal to this value. Default: 5,000 resources • CollectPoi.IncludeDeviations: Indicates whether deviations should be included in a POI collection. Default: False • CollectPoi.IgnoreChangedResourceSynchronization: Checks the configuration settings to see whether the changed resource synchronization should be ignored. Default: False
JobTicketManager	Is used for managing jobs between the different Data Governance Edition internal services. This is purely internal framework and there are no configuration parameters.
ManagedDomains	Provides an interface for managed domain information. This includes creating, querying and deleting managed domains, as well as validating service account access within a given domain. This service also maintains a cache of managed domain information which includes the service account. Every three minutes this information is refreshed.
ManagedHosts.InternalService	Provides managed host functionality for creating, updating, reinstalling and removing managed hosts. In addition, the service provides a framework for retrieving information about synchronized accounts, synchronized machines, synchronized SharePoint farms, and service accounts. This service also provides functionality for retrieving, upgrading, restarting, adding, removing, registering, unregistering, leasing and updating agents, as well as retrieving agent logs and parsing agent metrics.

Internal service name	Task description
ManagedResourceService	Exposes managed resource objects from the database layer. This includes creating, deleting, retrieving and updating managed resource types, managed group templates, group permissions, managed share root paths, managed resource domains, and name pattern resolvers. This service also provides information about managed resources and their relationship with data under governance. NOTE: The only public endpoints are PowerShell scripts for accessing and manipulating data for group templates. There are no internal processing and there are no configuration parameters used within this service.
ManagementServer	Manages the core Data Governance Edition dependencies, by ensuring a valid database connection is established, updating deployment information, creating and maintaining Data Governance Edition's service connection point, and maintaining deployment information, such as server version. **Configuration setting**: • MinimumSupportedModuleMigrationVersion: The minimum supported module migration version. If during installation, the new version is less than this value, the installation cannot occur. Default: Null **Registry settings**: • One Identity Manager database connection string (Q1IMDBConnectionString): The connection string for the One Identity Manager database. • Oracle deployment: One Identity Manager database (Q1IMDBPlatformOracle): Indicates whether the One Identity Manager database is an Oracle connection. Default: False (SQL connection) NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use. • Default identity SID (DefaultEmployeeSid): The SID of the default identity to be used by the Data Governance Edition topology crawler. Default:

Internal service name	Task description
	None - Resource Activity database connection string (QAMAuditActivityDBConnectionString): The connection string for the Data Governance Edition Resource Activity database. - Oracle deployment: Resource Activity database (QDGDBPlatformOracle): Indicates whether the Resource Activity database is an Oracle connection. Default: False (SQL connection). NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.
Metrics	Provides the framework for metric collection. Core metrics include POI metrics, agent communication metrics, and agent performance metrics. The frequency of metric collection is set using an entry in the application configuration file. **Configuration setting**: - Metrics.CollectionIntervalInSeconds: The interval at which metrics are collected. Default: 60 seconds.
ResourceActivityInternalService	Provides functionality related to resource activity and resource ownership. Actions include retrieving resource and trustee activity, calculating and granting perceived ownership, and aggregating resource activity. This internal task runs a synchronization every five minutes, which is not configurable. The task checks for "stale" entries in the QAMDuG table every five minutes after the Data Governance service starts. The LastOwnerShipCalculation column in the QAMDuG table stores the last time the synchronization ran. An entry is considered "stale" if one of the following is found to be true: - The LastOwnerShipCalculation column is empty (null). -OR- - The LastOwnerShipCalculation value is older than 24 hours (configurable in PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes setting

Internal service name	Task description
	in the Data Governance service configuration file). This service updates the perceived owner and POIs for governed resources on a regular interval, configurable within the application configuration file. **Configuration setting**: - PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes: The interval at which the perceived ownership calculation is updated and refreshed. Default: 1440 minutes (1 day). - PerceivedOwnershipActivityPeriod: The time period (in days) to look for past resource activity in order to determine perceived owners. Default: 30 days.
ResourceEnumeration	Provides functionality related to resource expansion, governance and publication. Actions include placing and removing resources under governance, publishing and unpublishing resources to the IT Shop, performing resource searches, and performing resource enumeration. All actions requiring service account credentials are performed using the server account for the targeted managed domain.
ResourcePolicyManagementService	Exposes resource policy objects from the database layer and provides the framework for resource provisioning. This includes the ability to create, delete, query and update access templates, trustee templates and resource policies. In addition, this service allows for resource provisioning. This internal service is for development purposes only.
ServerUpdatesService	Handles the updating of managed host states. For a description of managed host states, see the *One Identity Manager Data Governance Edition User Guide*.
ServiceAccounts	Handles actions regarding the Data Governance Edition service accounts. Actions include querying, creating, removing and validating service account credentials, and granting log on as a service rights to a given account. This service is consumed by both PowerShell and the Manager.

Manually deploying Data Governance service

You need the following to manually deploy the Data Governance service:

- Data Governance Server installation msi
- Local Administrator rights on the server where the Data Governance service is to be installed.
- Installation of the One Identity Manager client applications (including the Data Governance Edition PowerShell snap-in)
- Ability to change One Identity Manager configuration options in the Designer application
- Connection information to the One Identity Manager database
- Database creation permissions (if creating the Data Governance Resource Activity database)

To manually deploy the Data Governance service

1. Log on to the system with the One Identity Manager client installation.
2. Open the Designer and log on as a system user with administrative privileges (for example, viadmin)
3. Edit the Data Governance service configuration parameters:
   a. In the navigation view, select **Base Data** | **General** | **Configuration parameters**.
   b. In the far right column, click **Edit configuration parameters**.
   c. Expand **TargetSystem** | **ADS** | **QAM** | **QAMServer**.
   d. Change the **ServerName** value to the fully qualified DNS name of the server where the Data Governance service is to be installed.
   e. Set the **Port** value to the net.tcp port your server will listen on. The HTTP port will automatically be configured to use the net.tcp port value -1.
   f. Set the **Deployment** value to the name of your Data Governance Edition deployment.

      > NOTE: This defaults to "DEFAULT". If you are going to or already have multiple Data Governance Edition deployments in your Active Directory forest, you must ensure this name is unique. The Deployment value is restricted to a maximum of 30 characters and can contain alphanumeric characters and underscores (no spaces).
4. Use the LocalSystem account to log on to the Data Governance server specified above and run the Data Governance Server installation msi.

> NOTE: When you run the MSIEXEC from a command prompt, you must be running as local system. This ensures that the service connection point can be updated no matter what your Data Governance service runs as.

Example: ->msiexec /i "DataGovernance_ServerComponentsInstaller_x64.msi" /lv C:\DgeMsintallLog QAMDEPLOYMENT="testNew" QAMPORT=8722

> NOTE: Refer to the Microsoft documentation for command line syntax of MSIEXEC.EXE. For more information on using the Windows Installer (MSIEXEC.exe) refer to Microsoft's MSDN library: https://msdn.microsoft.com/en-us/library/aa367988(VS.85).aspx
>
> See Data Governance service options for a description of the Data Governance deployment options available.

5. Open a Windows PowerShell console on the machine with the One Identity Manager client installation.
6. Run the following cmdlet to import the Data Governance Edition PowerShell module:

   Import Module "<*path*>"

   Where <*path*> is the file path for the QAM.Client.PowerShell.dll assembly. By default, the <path> for the Data Governance server machine would be "C\:Program Files\One Identity\One Identity Manager\QAM.Client.PowerShell.dll"
7. Run the following PowerShell cmdlet to set the server name, deployment name and port information used by the Data Governance Edition commands to connect to the Data Governance server:

   Set-QServiceConnection -ServerName "<*DGE server machine name*>" -Port <*Value*> - Deployment "<*Deployment name*>"

   > NOTE: The <*DGE server machine name*>, Port <*Value*> and <*Deployment name*> must be the same values as specified in step 3.
8. Close the PowerShell console and restart the Data Governance service.
9. Run the following PowerShell cmdlet to establish the database connection between One Identity Manager and Data Governance Edition:

   Initialize-QDataGovernanceServer -DatabaseConnectionString "<*Connection string for Identity Manager database*>" [-DefaultEmployeeSid "<*SID of user account*>"

   > NOTE: Only specify the "-DefaultEmployeeSid" parameter if you want to take advantage of the automatic forest topology harvest. Adding this parameter adds the user associated with the specified SID to the One Identity Manager Identities with the appropriate Data Governance application roles. This provides the same functionality as selecting the **Add the current user to the One Identity Manager Identities with Data Governance application role**s option when using the Data Governance Configuration wizard.

   > NOTE: If Windows Integrated Authentication is used to connect to the database, the Data Governance server must be configured to run as an identity other than LocalSystem (See step 4).

Connection string examples:

An example of a connection string for Windows authentication may look like this:

> "Server=myServerAddress;Database=myDatabase'UserId=myUser;Password=myPassword;Trusted_Connection=True"

An example of a connection string for SQL authentication may look like this:

> "Data Source=myServerAddress;Intitial Catalog=myDatabase;User Id=myUser;Password=myPassword"

For more information on connection strings, see The Connection String Reference.

10. Using your preferred database management tool, browse on the Data Governance server to the %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server\Activity Database Scripts folder and locate a file named "DGAuditDatabaseCreationScript.sql".
11. For SQL Server hosted databases, open the DCAuditDatabaseCreationScript.sql file and update the database name specified in the CREATE DATABASE and USE statements.

    NOTE: If you are running multiple Data Governance Edition deployments, it is highly recommended that you append the deployment name to the database name (for example, DGE_DEFAULT). This database name has a maximum length of 30 characters and can contain alphanumeric characters and underscores (no spaces).
12. Run the appropriate script for your database management system to create the Data Governance Resource Activity database.
13. Run the following PowerShell cmdlet to initialize the database to store data generated when a managed host has resource activity collection enabled:

    > Initialize-QDataGovernanceActivity -ConnectionString "<*Connection string to activity database*>"

    NOTE: Ensure the connection string's Initial Catalog value (Database value if using Windows authentication) matches the name you specifies in the sql script when creating the Data Governance Resource Activity database.
14. Restart the Data Governance service.

    NOTE: It might take a few minutes before the Data Governance topology harvest task begins.

Data Governance service options

The Data Governance service installer is included in the autorun and can be found in the QAM module's directory. For example, C:\<*DGE Build*>\Modules\QAM\dvd\DataGovernance_ServerComponentsInstaller_x64.msi.

Only a 64-bit version is available.

The following options are available when using the Windows Installer .msi to install the Data Governance service.

NOTE: Log on to the Data Governance server with an account with administrative access and run the Data Governance Server installation msi from the command line, providing the options as described below.

Table 5: Data Governance service command line options

Option	Description
INSTALLDIR="<*Installation Directory Path*>"	Use this option to specify the folder on the local system into which the Data Governance service is to be installed.
QAMDEPLOYMENT="<*DGE Deployment Name*>"	Use this option to specify a unique name for the Data Governance Edition deployment. The deployment name has a maximum length of 30 characters; and can only contain alphanumeric characters and underscores (no spaces allowed). If no deployment name is specified, the default deployment name of "DEFAULT" is used.
QAMPORT="<*port number*>"	Use this option to specify the net.tcp port to be opened on the Data Governance service. If no port is specified, the default port of 8722 will be used.
SERVICEACCOUNT="< *DOMAIN\UserName*>"	Use this option to specify the service account to be used to access the One Identity Manager database. This is required if you are using Windows authentication to access the database.
SERVICEACCOUNTPASSWORD="< *Password*>"	Use this option to specify the password associated with the service account. This is required if you are using Windows authentication to access the database.

4

Data Governance agents

The *One Identity Manager Data Governance Edition Deployment Guide* provides details on adding managed hosts and deploying Data Governance agents; the information provided here is intended to provide more information about this deployment process and the Data Governance agents.

- Agent deployment process
- Agent files
- Data Governance agent configuration

Agent deployment process

1. The Data Governance service pushes the "QRemoteExecutorService.exe" file onto the agent host under a hidden folder:

   \\AgentHost\admin$\Broadway\AgentManagement
2. The Remote Executor is started on the agent, determines the agent architecture, and sends the data back to the Data Governance service.
3. The Data Governance service pushes the correct agent installer to the same location as the Remote Executor.
4. The Remote Executor installs the agent to %ProgramFiles%\One Identity\One Identity Manager Data Governance\Agent Services by default.
   1. Local agents are named "DGE_<*DeploymentName*>_LocalHost"

      Example: DGE_DEFAULT_LocalHost
   2. Remote agents are named "DGE_<*DeploymentName*>_<*FQDN of managed host*>"

      Example: DGE_DEFAULT_flowernetapp_flowers_local
   3. SharePoint Farm agents are named "DGE_<*DeploymentName*>_Sharepoint"

      Example: DGE_DEFAULT_Sharepoint

> NOTE: For multi-agent SharePoint managed hosts, an number is appended to the end of the agent service name.
>
> Example: DGE_DEFAULT_Sharepoint_1, DGE_DEFAULT_Sharepoint_2, DGE_DEFAULT_Sharepoint_3, and so on.

d. SharePoint Online agents are named "DGE_<*DeploymentName*>_SharePointOnline_<*Office 365 Host*>"

Example: DGE_DEFAULT_SharePointOnline_DGEPROD.ONMICROSOFT.COM

e. OneDrive for Business agents are named "DGE_<*DeploymentName*>_OneDriveBusiness_<*Office 365 Host*>"

Example: DGE_DEFAULT_OneDriveBusiness_DGEPROD.ONMICROSOFT.COM

5. The Data Governance service grants the service account used for agent deployment a few local privileges, including:

   a. SE_SERVICE_LOGON_RIGHT

   b. SE_TCB_NAME

   c. SE_RESTORE_NAME

   d. SE_BACKUP_NAME

Agent files

This table lists the files created when a Data Governance agent is deployed. All files associated with each agent instance are located in subdirectories of the agent installation folder.

- Local agent files are stored in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DGE_<*DeploymentName*>_LocalHost
- Remote agent files are stored in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DGE_<*DeploymentName*>_<*FQDN of managed host*>
- SharePoint Farm agent files are stored in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DGE_<*DeploymentName*>_Sharepoint

  > NOTE: For multi-agent SharePoint managed hosts, an number is appended to the end of the directory name.
  >
  > Example: DGE_DEFAULT_Sharepoint_1, DGE_DEFAULT_Sharepoint_2, DGE_DEFAULT_Sharepoint_3, and so on.

- SharePoint Online agent files are stored in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DGE_<*DeploymentName*>_SharePointOnline_<*Office 365 Host*>
- OneDrive for Business agent files are stored in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DGE_<*DeploymentName*>_OneDriveBusiness_<*Office 365 Host*>

Table 6: Agent files

File name	File type	Purpose
DataGovernance.Agent.exe.dlog	Trace Log Document	Agent log file. Double-click to display the Log Viewer to view the log.
dlog.config	XML document	Configuration settings for the agent log file.
server.config.xml	XML document	Current agent configuration settings from the Data Governance server. This file is an output of the configuration from the Data Governance server. It is overwritten upon each configuration from the server. NOTE: Do not edit this file.
.sqlite	SQLite file	SQLite database files are used for temporarily storing resource access, security and if enabled, resource activity: • ResourceAccessSync_*: Keeps track of what the agent has already synchronized with the Data Governance server. • ResourceActivityStore_*: Stores activity data for various host types. • ResourceSecurityStore_*: Stores scan data for various host types: SharePoint, NTFS, NFS and Cloud. • ResourceSecurityStore_Service Identities: Stores scan data for service logon accounts for Windows hosts. • ResourceSecurityStore_ WindowsComputer: Stores shares, local users and groups, and local rights for hosts which have an Active Directory computer object and for SharePoint.

File name	File type	Purpose
		NOTE: All of the *.sqlite* files are maintained by the agent process and are required for proper functionality. Do not attempt to view, edit, rename, move or delete any of these files.

In addition to the above mentioned agent files, the DataGovernance.Agent.exe.config file is stored in the Agent Services directory. This file contains agent configuration settings that cannot be applied using the Manager. Any changes made to this configuration file will apply to all agent instances running on the host. For more information on the agent configuration settings that can be changed, see Data Governance agent configuration file settings on page 91

Data Governance agent configuration

Data Governance agent configuration values are stored in one of the following places:

- Agents receive settings from the Data Governance service, and these settings can be viewed in the server.config.xml file in the agent instance folder under the Agent Services directory in the agent's installation directory: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\<*Agent instance*>.

  > NOTE: This file is an output of the configuration from the Data Governance server and is overwritten upon each configuration from the server. Do NOT edit this file.

- All agents on a managed host also contain settings stored in the DataGovernance.Agent.exe.config file in the Agent Services directory in the agent's installation directory: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services.

  > NOTE: The DataGovernance.Agent.exe.config file only contains settings that are NOT available through the Manager. Any changes made to this configuration file will apply to all agent instances running on the host. For more information on the Data Governance agent configuration file settings that can be configured in the DataGovernance.Agent.exe config file, see Configurable configuration file settings.

> NOTE: With the new agent architecture implemented in Data Governance Edition version 7.0.2, the DataGovernance.Agent.exe.config file contains a subset of the settings provided in the legacy agent. This is because the Data Governance server configuration is applied BEFORE this file. So this file now only contains settings that are NOT available thorough the Manager to ensure agent defaults are not overridden. Also, the legacy agent registry key settings are no longer available for configuration purposes.

5

Resource activity collection in Data Governance Edition

Resource activity collection recap:

- Collecting resource activity is supported for local managed Windows servers, SharePoint farms, and supported NetApp and EMC managed hosts. Resource activity collection is not supported for Windows Cluster/Remote Windows Computer, Generic or Cloud managed hosts.
- Collects data for resources in the folders that are specified on the **Managed Paths** page of the **Managed Hosts Settings** dialog.
- Collects data on identities, security changes, creates, deletes, renames, writes, and reads on resources.

  > NOTE: Read operations are disabled by default for all managed hosts. To enable read operations on a managed host:
  >
  > 1. Open the Manager.
  > 2. In the Navigation view, select **Data Governance | Managed hosts**.
  > 3. Select the required managed host from the **Managed hosts** view.
  > 4. Select **Edit host settings** from the Tasks view or right-click menu.
  > 5. Open the **Resource Activity** page.
  > 6. Select the **Read** check box.
  > 7. Click **OK** to save your selections and close the **Managed Host Settings** dialog.

- Data Governance Edition is NOT an auditing tool:
  - It captures the account who performed the action.
  - It does NOT capture where the action was generated from (for example, IP Address).
  - It does NOT store the "from" and "to" values; only that a certain action was performed on some resource by someone.
  - It does NOT store the exact times the action was performed.

> NOTE: Activity is stored in "time spans". Aggregation levels control how much data is stored.
>
> For example, Bill opens a spreadsheet on a file server at 1:05 pm. He saves it five times in the next 45 minutes and then closes it. The aggregation level for managed host is set to one hour. When the aggregation window closes, there will be three entries sent to the Resource Activity database:
>
> - One entry for the "open" action
> - One entry for the "save" action (with a count of 5)
> - One entry for the "close" action
>
> The entries will show that the action occurred between 1:00 pm to 2:00 pm, but there will be no indication of when specifically within that hour the action took place.

- Resource activity collection and aggregation is disabled by default and can be enabled on a per-managed host basis using the **Resource Activity** page on the **Managed Host Settings** dialog.
- When resource activity collection is enabled, certain well-known system accounts, file extensions, and folders are excluded by default. For each managed hosts, you can modify what is excluded from resource activity collection using the **Resource Activity** page on the **Managed Host Settings** dialog.

  > NOTE: The agent will always filter out activity generated by the agent service account regardless if the service account is specified in the Resource Activity Exclusions. This applies to all local and remote managed hosts; however, the agent service account for SharePoint managed hosts are not excluded by default. You will need to add the SharePoint service account manually for SharePoint managed hosts.

- Aggregated activity data forwarded by the Data Governance agents or harvested from Change Auditor is stored in a central database, Data Governance Resource Activity database. Only the Data Governance service interacts with this database.
- The Data Governance server periodically retrieves resource activity summary information to calculate perceived ownership suggestions for resources under governance.
- If you are collecting resource activity, set up a scheduled execution of the activity database compression utility to ensure your Resource Activity database remains manageable. For more information, see Scheduling activity compression and deletion on page 42.
- Reports that use resource activity information include:
  - Resource Activity
  - Account Activity
  - Interesting Resources without an Owner
  - Data Owners vs. Perceived Owners
  - Perceived Owners for Data Under Governance

Resource Activity database maintenance

The Resource Activity database stores resource activity information. To ensure that activity data remains manageable and usable, you need to control the growth of activity in this database. Data Governance Edition provides the following ways to control the size of the Resource Activity database:

- For managed hosts that are tracking resource activity, you can exclude selected accounts, file extensions, and folders to be scanned by agents depending on the type of managed host.
- Use the resource activity deletion and compression utilities to manage the growth of your database.
  - It is recommended that you set up a schedule to run the activity database compression utility, which compresses the activity in your database that is older than a certain age and optionally purges entries that are even older. For more information, see Scheduling activity compression and deletion on page 42.
  - You can manually run the activity compression utility to compress activity from multiple rows into a single row in the database. For more information, see Manually running the activity compression utility on page 43.
  - You can manually run the activity deletion utility to remove activities that are no longer needed. For more information, see Manually running the activity deletion utility on page 46.
- For a given managed host, the Clear-QResourceActivity PowerShell cmdlet enables you to remove activity data from the database on demand when it is no longer required. For more information, see Clear-QResourceActivity on page 159.

Configuring Resource Activity database maintenance

The resource activity deletion and compression utilities provided with Data Governance Edition can help you manage the growth of your database. Settings for the activity compression and deletion utilities can be set in the Data Governance server configuration file, DataGovernanceEdition.Service.exe.config, which is located in the Data Governance server directory.

Enabling and setting up activity database maintenance here in the configuration file ensures that activity is compressed and deleted on a schedule.

The section in the configuration file that controls deletion/compression for activity is as follows:

```
<!--
```

```
Activity compression configuration.
    enabled: Indicates whether the scheduled compression is enabled or not.
    dailyExecutionTime: Time of day to perform the compression. Format is (h:m:s:[z]).
    If the time zone [z] is not specified universal time is assumed.
    compressOlderThan: The default lower bound for activity data compression. Any
    activity data older than this value will be compressed.
    deleteOlderThan: The lower bound for activity data deletion. Any activity data older
    than this value will be deleted. Do not specify this value if deletion is not desired.
    deletionBatchSize: The batch size used during deletion.
-->
    <activityCompressionConfiguration enabled="false" dailyExecutionTime="23:00:0"
    compressOlderThan="0d" deleteOlderThan="180d" deletionBatchSize="5000">
<!--
    NOTE: These values are not to be changed without the assistance of support
    personnel.
    Defines the activity compression passes to perform.
    compressOlderThan: The lower bound for activity data compression for a given pass.
    Any activity data older than this value will be compressed. If this value is not
    specified the default is used.
    aggregationPeriod: The aggregation period to use in a given pass.
-->
<passes>
    <add aggregationPeriod="1h"/>
    <add aggregationPeriod="1d"/>
    <add aggregationPeriod="30d"/>
    <add compressOlderThan="180d" aggregationPeriod="180d"/>
    <add compressOlderThan="720d" aggregationPeriod="360d"/>
</passes>
</activityCompressionConfiguration>
<!-- Application settings -->
```

Scheduling activity compression and deletion

To ensure that activity data remains manageable and usable, it is recommended that you schedule the Data Governance service to compress or delete activity data once per day. The scheduled compression process aggregates similar activity entries within a given time span into one entry.

NOTE: Once activity data has been compressed, it cannot be uncompressed.

To schedule activity compression

1. Browse to the DataGovernanceEdition.Service.exe.config file (located in the Data Governance Service installation directory).
2. Open the file and navigate to the xml node ‘activityCompressionConfiguration’.

   The default values are as follows:

   ```
   <activityCompressionConfiguration
   enabled="false"
   dailyExecutionTime="23:00:0"
   compressOlderThan="0d"
   deleteOlderThan="180d"
   deletionBatchSize="5000">
   ```
3. Enable the scheduled compression, by changing the value of the ‘enabled’ attribute to true.
4. Set the daily execution time through the ‘dailyExecutionTime’ attribute.

   The valid format is ‘h:m:s:z’, where ‘h’ is hours, ‘m’ is minutes, ‘s’ is seconds and ‘z’ is the optional time zone offset value. If the time zone [z] is not specified, universal time is assumed.
5. Configure which data to compress through the ‘compressOlderThan’ attribute. Any activity data older than this value is compressed.
6. Configure which data to delete through the ‘deleteOlderThan’ attribute. Any activity data older than this value is deleted. No deletion occurs if this value is not specified.
7. Configure the maximum number of rows to delete at a time through the 'deletionBatchSize' attribute.
8. Save your selections.

Manually running the activity compression utility

The Activity Compression utility can be used to further manage the database. It enables the compression of activity from multiple rows into a single row in the database. This utility is located in the Data Governance Server installation directory, %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server\ActivityCompression.exe.

Best practices

Before using this utility:

- Backup the activity database and transaction log. This clears completed transactions from the transaction log file to free space for new transactions generated by running the utility. For additional information, refer to Books Online for SQL Server - Transaction log backups.

> NOTE: The utility will change the SQL Server recovery model to bulk-logged model. This ensures that the transaction log will not grow too large, using up available disk space, while the utility is running.

While running the utility:

- The utility can compress activity data while managed hosts, with resource activity tracking enabled, continue to monitor changes. However, it will run faster if resource activity tracking is disabled on all managed hosts.

  When you specify a large compression granularity (such as 1 year) the utility may use a large percentage of the SQL Server's resources and there is a risk of the SQL commands timing out. To prevent this, stage the compression by first running with a compression granularity of 1 week and then running the utility again specifying a compression granularity of 1 year.

Once you have run the utility:

- If bulk-logged recovery model is not suitable for your recovery model, change the database recovery model to either simple or full recovery model. See Books Online for SQL Server – Change recovery model.

  If the database size and log size are still too large, backup the database again to clear completed transactions from the log file and then execute “DBCC SHRINKDATABASE” on the database to shrink the database and log to the correct size. For more information, see Books Online for SQL Server – DBCC SHRINKDATABASE.

Installation

The ActivityCompression utility is a console program.

To install the utility

1. Copy the ActivityCompression utility to the database server or a member server.
2. Open a command prompt and enter ActivityCompression to see usage instructions.

Using the utility

```
ActivityCompression
[-ConnectionString] <"String">
[[-RemainUncompressed] <integer>]
[[-CompressionGranularity] <integer>]
[[-DatabasePlatformOracle] <string>]
```

Where:

-ConnectionString: The string used to connect to the database.

- SQL Server Authentication: -ConnectionString "Data Source=myServerAddress; Initial Catalog=myDatabase; User Id=myUser; Password=myPassword"
- Windows Authentication: -ConnectionString "Server=myServerAddress;Database=myDatabase;Trusted_Connection=True"

-RemainUncompressed: The amount of time (in days) that will remain uncompressed. By default, this utility will compress all but the 7 days from the current date time. This allows the newest activity in the database to remain uncompressed for more in depth analysis.

Example: -RemainUncompressed 7

-CompressionGranularity: The amount of time (in days, default 7 days) to compress the fine grain activity entries to. If your database is configured with daily activity granularity, setting -CompressionGranularity to 7 will compress all activity within those seven existing timespans into a single timespan with the similar activity over the seven days compressed into single records.

Example: -CompressionGranularity 7

-DatabasePlatformOracle: The database platform used for the activity database. If not specified, this defaults to SQL Server.

- SQL Server platform: -DatabasePlatformOracle false
- Oracle platform: -DatabasePlatformOracle true

> NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

Examples

Compress all activity to a 7 day granularity

-ConnectionString "Server=myServerAddress;Database=myDatabase; Trusted_Connection=True" -RemainUncompressed 0 -CompressionGranularity 7

Compress all but the 30 days from current time to a 7 day granularity

-ConnectionString "Server=myServerAddress;Database=myDatabase;Trusted_Connection=True" -RemainUncompressed 10 -CompressionGranularity 7

Notes on running the utility

- The first compression is the most resource intensive. Once the initial compression is completed, future scheduled compressions take a fraction of the time to complete.
- As the number of managed hosts increases, the processing time per record will also increase. This may cause large deployment of 1000+ managed hosts to take a week or more to complete the initial compression.
- The recommended maximum “CompressionGranularity” on the first compression is seven days. This number may be higher depending on the specifications of the SQL Server.
- You can run the compression utility in a production environment. However, we recommend scheduling this process to run on a weekly basis.

- You may run the compression multiple times to create differing levels of detail the further back in time you go.

  Assume a batch file running the compression three times with the following settings:
  -CompressionGranularity 7 –RemainUncompressed 7
  -CompressionGranularity 30 –RemainUncompressed 30
  -CompressionGranularity 365 –RemainUncompressed 365

  The first run will compress the activity into week long blocks, keeping the most recent week uncompressed. The second run will then further compress the database, compressing activity older than 30 days into month long blocks. The final compression run will compress anything older than 365 days into year time blocks. This will ensure that the most recent activity can be examined at a very detailed level while summing older activity to keep the database size reasonable.
- If you experience timeouts, reduce the CompressionGranularity to one day. Compress further from there. This timeout does not cause any data issues, but it does interrupt the compression and it will need to be run again.

Manually running the activity deletion utility

You can manually run the Activity Deletion utility to remove activities that are no longer useful and are using hard drive space. This utility is located in the Data Governance Server installation directory, %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server\ActivityDeletion.exe.

Best practices

Before using this utility ensure that you:

- Back up the activity database and transaction log. This clears completed transactions from the transaction log file to free space for new transactions generated by the utility. For additional information, see Books Online for SQL Server — Transaction log backups.

> NOTE: The utility changes the SQL Server recovery model to bulk-logged model. This ensures that the transaction log does not grow too large, using up available disk space, while the utility is running.

While running the utility:

- The utility can delete activity data while managed hosts have activity tracking turned on; however, it is recommended to turn it off while the utility runs to increase performance.

  If activity is turned off, we recommend increasing the number of rows to delete from the tables to 100 K. You can increase the batch size by specifying –ActivityDeleteBatchSize 100000 on the command line.

If activity tracking is turned on or you have a database server that is not dedicated as the resource activity database, use the default batch size (5k) or less to avoid having the utility consume all the database servers resources.

Once you have run the utility:

- If bulk-logged recovery model is not suitable for your recovery model, change the database recovery model to either simple or full recovery model. See Books Online for SQL Server — Change recovery model.

Installation

The ActivityDeletion utility is a console program.

To install the utility

1. Copy the ActivityDeletion utility to the database server or a member server with .NET 4.5.1 or lower installed.
2. Open a command prompt and enter ActivityDeletion to see usage instructions.

Using the ActivityDeletion utility

> NOTE: Ensure that you back up your database before running the utility.
>
> Ensure that the utility has finished processing before running a new instance.

ActivityDeletion

[-ConnectionString] <"String">

[[-DaysOfActivityToKeep] <integer>]

[[-ActivityDeleteBatchSize] <integer>]

[[-DatabasePlatformOracle] <string>]

Where:

-ConnectionString: The string used to connect to the database.

- SQL Server Authentication: -ConnectionString "Data Source=myServerAddress; Initial Catalog=myDatabase; User Id=myUser; Password=myPassword"
- Windows Authentication: -ConnectionString "Server=myServerAddress;Database=myDatabase;Trusted_Connection=True"

-DaysOfActivityToKeep: The amount of time (in days) that remains undeleted. By default, this utility deletes all but 60 days from the current date time.

Example: -DaysOfActivityToKeep 30

-ActivityDeleteBatchSize: The number of rows to delete (at one time) from the database's activity tables. If you have chosen to turn off resource activity tracking while this utility deletes data, then a batch size of 100000 is recommended; otherwise, use the default of 5000.

Example: -ActivityDeleteBatchSize 10000

-DatabasePlatformOracle: The database platform used for the resource activity database. If not specified, this defaults to SQL Server.

- SQL Server platform: -DatabasePlatformOracle false
- Oracle platform: -DatabasePlatformOracle true

  > NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

Example

Resource activity tracking is turned off; delete all activity older than 30 days:

-ConnectionString "Server=myServerAddress;Database=myDatabase; Trusted_Connection=True" –DaysOfActivityToKeep 30 –ActivityDeleteBatchSize 100000

Troubleshooting

A log file named "ActivityDeletionLog.txt" is generated in the location where the ActivityDeletion utility was started.

The ActivityDeletionLog shows the progress of the utility and any errors it encounters.

Verifying resource activity is making it to the Resource Activity database

There are a number of ways, as described below, to verify that resource activity is being recorded properly:

- At the agent level, on the agent host, in the agent instance directory, you can watch the "ResourceActivityStore_XYZ.sqlite" file increase in size.
- For each aggregation interval, observe the creation of the ResourceActivityStore_*retired file. These files contain activity that will be forwarded to the Data Governance server.
- At the Data Governance server level, check the DataGovernanceEdition.Service.exe.dlog file for a message similar to the one below which is logged when an agent sends activity to the Data Governance server (search for the words in bold):

  ```
  2016-07-2015:14:33:539 [16][INFO][SendSplitMessageResponses(179)] Sending
  UpdateResourceUsage in 1 parts.
  ```
- In the Manager, compare the following agent statics in the **Agents** view:
  - Activity Enqueued: The number of resource activity records that have been queued and are waiting to get stored/aggregated in the Resource Activity store.

  - Activity Processed: The number of resource activity records that have been processed and stored in the Resource Activity store.
- In the Manager, run the Resource Activity report:
  - In the Resource browser or Governed data overview, locate the target resource.
  - Select the target resource and select **Resource activity report**.
  - Specify the appropriate time range and click **Finish** to generate the report.
  - If the report lists the expected activities, activity is being correctly recorded.
- In the Data Governance Edition Resource Activity database, check if there is any items in dbo.AuditUsage. If there is, activity is correctly being sent from the agent to the Data Governance server and then to the Data Governance Edition Resource Activity database.

Cloud managed hosts permission level to role mapping

For cloud managed hosts, Data Governance Edition assigns and displays a role instead of individual permission levels in the Resource Browser and Resource Access report. In addition, these roles are used when security information is used to calculate perceived ownership of a resource. The following tables explain the cloud managed host permission level to role mapping used in Data Governance Edition.

Default permission levels

The default permission levels are mapped to roles in the following manner.

Table 7: Default permission level mapping

Permission level	Role
Full Control	owner
Design	writer
Edit	writer
Contributor	writer
Read	reader

Custom permission levels

For custom permission levels, the underlying permissions are analyzed and the highest role is assigned as described in the following tables.

Table 8: Custom permission level mapping: List permissions

List permission	Role
Manage Lists	writer
Override List Behaviors	writer

List permission	Role
Add Items	writer
Edit Items	writer
Delete Items	writer
View Items	reader
Approve Items	writer
Open Items	reader
View Versions	reader
Delete Versions	writer
Create Alerts	writer
View Application Pages	reader

Table 9: Custom permission level mapping: Site permissions

Site permission	Role
Manage Permissions	writer
View Web Analytics Data	reader
Create Subsites	writer
Manage Web Site	writer
Add and Customize Pages	writer
Apply Themes and Borders	writer
Apply Style Sheets	writer
Create Groups	writer
Browse Directories	reader
Use Self-Service Site Creation	writer
View Pages	reader
Enumerate Permissions	reader
Browse User Information	reader
Manage Alerts	writer
Use Remote Interfaces	writer
Use Client Integration Features	writer
Open	reader
Edit Personal User Information	writer

Table 10: Custom permission level mapping: Personal permissions

Personal permissions	Role
Manage Personal View	writer
Add/Remove Personal Web Parts	writer
Update Personal Web Parts	writer

QAM module tables

Data Governance Edition information is stored in the QAM module tables in One Identity Manager. This chapter provides some additional details regarding some of the commonly used QAM module components.

- QAM tables
- QAM views
- Resource types
- Trustee types

QAM tables

The following One Identity Manager database tables are used to store Data Governance Edition data.

Table 11: QAM module: Tables

Table name	Description
QAMAgent	Contains the installed agents for all locally managed hosts and remote hosts. Includes the correlation to a managed host, current agent status, agent version, agent name and public key information. Example: Agent DGE-SERVER is a local agent monitoring the server DGE-SERVER. Current status is OK and current version is x.x.
QAMAgentEvent	Stores the critical errors sent in by a running agent. You can view or clear critical errors through the **Agents** view in the Manager.
QAMAgentRoot	Contains the managed paths for all installed agents.

Table name	Description
	Contains the responsible agent, the full path of the root, and the root type. This information is pushed to the agent configuration file as well. Example: \\dge-server\C$\Shares\Share1 is a folder managed path for agent DGE-SERVER.
QAMClassificationLevel	Stores data about the classification levels (pre-defined or customer-defined) available for classifying data.
QAMDfsTarget	Contains the DFS paths for all managed DFS hosts. Includes information pertaining to DFS targets, associating local paths on a given server to a DFS managed host: Local Path, Target Server, Target Share, DFS Path and DFS managed host. Example: DFS-Folder is a DFS target located on server X at local path Y associated with DFS managed host Z.
QAMDuG	Contains the resources under governance across all managed hosts, including the responsible managed host, resource type, security descriptor, paths, business ownership information, as well as whether the data is a point of interest, is published to the IT Shop, is stale, or is a backing folder for a share. Example: Share1 is an NTFS/Folder resource that is a point of interest, currently published to the IT Shop using Folder security, and owned by Gary. Last point of interest calculation occurred 15 minutes ago.
QAMHelper*	These tables help correlate accounts found in permissions, and therefore in QAMTrustee, to their identity, synchronized by One Identity Manager. These tables are also used by the web portal to map accounts and identities used to calculate perceived owners. For example, it shows the correlation between an Active Directory user found in a security index on an agent to the Active Directory account synchronized within an Active Directory domain.
QAMLocalGroup	Stores the local groups discovered and synchronized on a Windows computer by the local agent.
QAMLocalUser	Stores the local users discovered and synchronized on a

Table name	Description
	Windows computer by the local agent.
QAMLocalUserInLocalGroup	Correlates the local user accounts in QAMLocalUsers with the groups they belong to in QAMLocalGroups.
QAMNode	Contains the installed managed hosts. The managed host information includes the host type, status, and agent configuration settings such as: file system activity settings, file system indexing settings, and file system scanning settings. Example: DGE-SERVER is a Windows Server, currently in OK status, with 256 total resources under governance, and 256 points of interest. The current agent configuration excludes x files and folders, synchronizes activity every 15 minutes under a five minute aggregation, and scans security index information once a day.
QAMOtherSIDInLocalGroup	Stores Active Directory accounts found in local groups by a local agent that were not resolved in Active Directory. This links to Active Directory sync of unresolved SIDs.
QAMScannerInfo	Stores the agent scanner states. For example, a scanner would be the Windows Computer, Service Identities, Local Groups, NTFS, SharePoint, NFS, and Cloud. Each of these "scans" the managed paths collecting security data.
QAMSecurityIndex	Contains direct access points for accounts that have been scanned by Data Governance agents, indicating the type of access that they have. Examples: • Matt has folder access on Windows Server A according to Agent X • Rita has share access on Windows Server B according to Agent Y
QAMTrustee	Contains information for security accounts that have explicit ACL security. This table is closely paired with QAMSecurityIndex and contains the specific account information, such as the account's security identifier (SID). Example: Gary with SID 123, is a Domain User, and has a display value of Domain\Gary.

QAM views

The following One Identity Manager views (queries) retrieve Data Governance Edition resource activity and security information.

Table 12: QAM module: Views

View name	Description
QAMResourceActivitySummary	Contains a summary view of who has generated activity events on what resources. The summary contains information about the trustee account, the managed host, and the activity the account generated on the resource. Example: Gary performed a delete operation on governed resource X located on managed host Y.
QAMResouceSecuritySummary	Contains a summary view of who has what security permissions on what resources. The summary contains information about the trustee account, the resource under governance, the managed host, and the access information that the account has on the resource. Example: Gary has AllowFullControlAccess on governed resource X located on managed host Y.

Resource types

The following resource types are referenced in Data Governance Edition data.

Table 13: Resource types

Value	Resource type
0101	Windows Computer\Share
0102	Windows Computer\Local User Rights
0103	Windows Computer\Operating System Administrative Rights
0201	NTFS\File
0202	NTFS\Folder

Value	Resource type
0301	Service Identities\Windows Service Identity
0401	SharePoint\Farm
0402	SharePoint\FarmAdminRight
0403	SharePoint\WebAppPolicy
0404	SharePoint\SiteCollectionAdminRight
0405	SharePoint\ServiceApplicationPermission
0406	SharePoint\ResourceItem
0407	SharePoint\WebApplication
0408	SharePoint\SiteCollection
0409	SharePoint\Site
0410	SharePoint\List
0411	SharePoint\Folder
0412	SharePoint\ListItem
0601	DFS\Link
0701	NFS\File
0702	NFS\Folder
0801	Cloud\File
0802	Cloud\Folder

Trustee types

This table lists the types of accounts that Data Governance Edition is aware of.

Table 14: Trustee types

Value	Trustee type
1	Domain User
2	Domain Group
3	Domain
4	Alias

Value	Trustee type
5	Wellknown
6	Deleted
7	Invalid
8	Unknown
9	Computer
60000	Broadway
60001	Machine Local User
60002	Machine Local Group
60003	SharePoint Identifying Claim
60004	SharePoint Group
60005	SharePoint Claim
60006	Unix Owner
60007	Unix Group
60008	Unix Other
70001	AzureAD User
70002	AzureAD Group
70003	SharePointOnline User
70004	SharePointOnline Group

8

Configurable configuration file settings

Data Governance Edition provides configuration files for the Data Governance service and the Data Governance agents.

- Data Governance service configuration file settings: This configuration file contains the server, self-service and points of interest (POI) configuration settings that can be modified.
- Data Governance agent configuration file settings: This configuration file contains the agent configuration settings that can not be set in the Manager client.

Data Governance service configuration file settings

The following Data Governance service configuration settings can be configured in the DataGovernanceEdition.Service.exe.config file in the server directory: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server.

Table 15: Server settings

Configuration setting	Description
AgentLeaseRenewPeriod	Sets the agent lease renewal interval.
DfsDataSyncInterval	Sets the default DFS synchronization interval.
DirectAccessForServiceCredentials	Allows the Data Governance service to access service account credentials directly from the database.

Configuration setting	Description
DisablePerceivedOwnershipUpdate	Can be used to disable the automatic perceived owner calculation for governed data.
EMCIsilonDFSLinksEnabled	Defines whether to enable the resolution of DFS links pointing to EMC Isilon CIFS device's folder.
EMCIsilonUseNetworkPathForAccessRequests	Defines whether to enable reading of access groups and their associated permissions directly from network share for resource access requests for DFS links.
FolderSecurity.UseAdminPathsForShareFolders	Controls how the Data Governance server deals with the security that backs folders.
ManagedHostDeleteBatchSize	Defines the batch size used to delete managed hosts and their associated resources and resource activity records from the database.
MessagingCacheFolder	Defines the server messaging cache location.
Metrics.CollectionIntervalInSeconds	Sets the metrics collection interval.
MinimumSupportedModuleMigrationVersion	Specifies the minimum supported module migration version.
OracleBulkImportBatchSize	Specifies the number of records to be imported at a time during a bulk import for an Oracle database. NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.
PerceivedOwnershipActivityPeriod	Defines the time period (in days) to look for past

Configuration setting	Description
	resource activity to determine perceived owners.
PerceivedOwnershipByResourceActivity	Indicates the primary source for calculating perceived owners: resource activity history or security information.
PerceivedOwnershipByResourceOwner	Indicates whether the access control list owner within the target system should be considered as a perceived owner suggestion.
PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes	Sets the perceived ownership update interval.
PerceivedOwnershipMaxReturnValue	Defines the maximum number of perceived ownership suggestions returned as a result of calculating perceived owners for a resource.
RemoteExecutor.WaitResultTimeout	Defines how long the Data Governance service should wait for results from the RemoteExecutor before timing out.
RemoteHostForCloudOnlyImplementation	Specifies the DNS host name of the Windows Server to be used for deploying remote agent for cloud hosts.
RestServicePort	Sets the communication port for HTTP protocol and REST services. (Communications with PowerShell and One Identity Manager clients and web server.)
SuggestedAgentCap	Defines the suggested maximum number of agent instances on a given computer.
SyncDomainPasswordInterval	Sets the managed domain

Configuration setting	Description
	and security information cache refresh interval.
VerboseHostForTrusteeLogging	Debug setting used to log the complete Alias table used for the query.

Table 16: Self-service settings

Configuration setting	Description
SelfService.AllowNonPublishedGroups	Indicates whether groups not published to the IT Shop are displayed in self-service web portal.
SelfService.AllowUnsychronizedGroups	Indicates whether groups not synchronized with One Identity Manager are displayed in self-service web portal.
SelfService.EnableSelfServiceRequest	Indicates whether self-service requests are enabled.
SelfService.IncludeSuitabilityTraceInfo	Indicates whether the suitability trace information is to be included as a property in the self-service request results.
SelfService.MarkSuitabilityVisible	Indicates whether the suitability integer is to be shown in the user interface when self-service groups are returned.
SelfService.MaximumMethodsCount	Defines the maximum number of self-service groups that can be returned for consideration.
SelfService.SuitabilityThreshold	Specifies the lowest possible suitability score to be used when returning self-service groups.

Table 17: Points of interest (POI) settings

Configuration setting	Description
CollectPoi.CheckFrequencyInMinutes	Sets the stale POI information check interval.
CollectPoi.IgnoreChangedResourceSynchronization	Indicates whether the changed resource synchronization should be ignored.
CollectPoi.IncludeDeviations	Indicates whether deviations are to be included in POI query.
CollectPoi.MaxConcurrentQueries	Defines the maximum number of simultaneous POI queries to be

Configuration setting	Description
	performed.
CollectPoi.OverdueThresholdInMinutes	Sets the amount of time before a resource is considered to be overdue for POI collection.
CollectPoi.QueryBatchSize	Defines the threshold on which a query is sent to the agent.
CollectPoi.QueryTimeoutInMinutes	Sets the amount of time before a POI query expires.
CollectPoi.QueryUpperBound	Defines the maximum number of resources to be returned from a POI query.

Table 18: Custom host parameters

Configuration setting	Description
additionalOperatingSystems	Allows you to specify additional operating systems so that those hosts can be added as generic managed hosts

In addition to the server, POI collection, and self-service settings listed above, you will find the following settings in the Data Governance service configuration file:

- Activity compression utility and activity deletion utility configuration. For more information on enabling and configuring these database utilities, see Resource Activity database maintenance on page 41.
- Activity weight multipliers used for calculating a resource's perceived owner. For more information on the activity weight multipliers, see Activity weight multipliers on page 83. For more information on configuring the perceived owner calculation, see the *One Identity Manager Data Governance Edition User Guide*.
- Self-service suitability calculators for determining the "best fit" groups for providing resource access. For more information, see Self-service suitability calculation multipliers on page 84.

additionalOperatingSystems

This parameter allows you to specify additional operating systems so that those hosts can be added as generic managed hosts.

> NOTE: Generic Managed Host functionality is meant to allow for the scanning of SMB shares and subfolers that are hosted on servers on an Active Directory joined computer. In order to be a Generic Managed Host, the server must be synchronized into the ADSMachine table, with the ADSMachine.DNSHostName set.
>
> Generic Managed Host functionality is meant to allow for the scanning of SMB shares and subfolers that are hosted on servers on an Active Directory joined computer. In order to be a Generic Managed Host, the server must be synchronized into the ADSMachine table, with the ADSMachine.DNSHostName set.

Table 19: Configuration setting: additionalOperatingSystems

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<customHostParameters>

Setting	<customHostParameters>
	  <additionalOperatingSystems>
	    <!--<operatingSystem value="<MyOperatingSystem"/>-->
	  </additionalOperatingSystems>
	</customHostParameters

Value	When the `operatingSystem` line is left as is (as a comment), Data Governance Edition does not recognize unsupported host types and therefore they can not be added as a generic managed host. When the `operatingSystem` line is no longer commented out and you specify the operating system for the hosts you want to manage, they will appear as an **Unknown** host type in the **Managed host** view which can then be added as a generic managed host.

How to modify	If you do not see the host you want to manage listed in the **Managed host** view, edit this parameter as follows:
	• Remove the commented `operatingSystem` line and replace it with a line that specifies the operating system value for the host you want to manage. That is, the string found in the ADSMachine.OperatingSystem field. For example, if the host you want to manage has the operating system field "My OS", edit this setting as follows:
	`<customHostParameters>`
	  `<additionalOperatingSytems>`

```
    <operatingSystem value="My OS"/>
  </additionalOperatingSystems>
</customHostParameters>
```

This will include all machines that contain the string "My OS" in its operating system field.

- If you want to specify an exact match, include the `isExact` parameter as follows:

```
<customHostParameters>
  <additionalOperatingSytems>
    <operatingSystem value="My OS" isExact="true"/>
  </additionalOperatingSystems>
</customHostParameters>
```

All of the hosts found using this filter will now appear in the **Managed hosts** view as **Unknown** host type.

AgentLeaseRenewPeriod

This key defines the refresh rate (in minutes) at which the server checks for expired agent leases. This key is used by the AgentLeaseManager internal service that handles agent lease management.

Table 20: Configuration setting: AgentLeaseRenewPeriod

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="AgentLeaseRenewPeriod" value="5"/>
Value	Default: 5 minutes
How to modify	Replace the value as required.

CollectPoi.CheckFrequencyInMinutes

This key defines the frequency (in minutes) at which the server checks for stale points of interest (POI) information. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve POI information on governed resources.

Table 21: Configuration setting: CollectPoi.CheckFrequencyInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.CheckFrequencyInMinutes" value="10"/>
Value	Default: 10 minutes
How to modify	Replace the value as required.

CollectPoi.IgnoreChangedResourceSynchronization

This key checks the configuration settings to see whether the changed resource synchronization should be ignored. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve points of interest (POI) information on governed resources.

Table 22: Configuration setting: CollectPoi.IgnoreChangeResourcesSynchronization

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.IgnoreChangedResourceSynchronization" value="false"/>
Value	Valid values: • false: do not ignore the changed resource synchronization (default) • true: ignore the changed resource synchronization
How to modify	Replace the value as required.

CollectPoi.IncludeDeviations

This key determines whether to collect information for objects whose security deviates from their parent's. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve points of interest (POI) information on governed resources.

Table 23: Configuration setting: CollectPoi.IncludeDeviations

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.IncludeDeviations" value="false"/>
Value	Valid values: • false: do not include deviations in POI collection (default) • true: include deviations in POI collection
How to modify	Replace the value as required.

CollectPoi.MaxConcurrentQueries

This key defines the maximum number of simultaneous queries that can be performed. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve points of interest (POI) information on governed resources.

Table 24: Configuration setting: CollectPoi.MaxConcurrentQueries

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.MaxConcurrentQueries" value="5"/>
Value	Default: 5 queries
How to modify	Replace the value as required.

CollectPoi.OverdueThresholdInMinutes

This key defines the amount of time (in minutes) before a resource is considered to be overdue for point of interest (POI) collection and a POI collection is initiated. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve POI information on governed resources.

Table 25: Configuration setting: CollectPoi.OverdueThresholdInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.OverdueThresholdInMinutes" value="1440"/>
Value	Default: 1440 minutes (one day)
How to modify	Replace the value as required.

CollectPoi.QueryBatchSize

This key defines the maximum number of resources sent in a single query to an agent.

This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve points of interest (POI) information on governed resources.

Table 26: Configuration setting: CollectPoi.QueryBatchSize

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.QueryBatchSize" value="5000"/>
Value	Default: 5000 resources
How to modify	Replace the value as required.

CollectPoi.QueryTimeoutInMinutes

This key defines the amount of time (in minutes) before a point of interest (POI) query expires. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve POI information on governed resources.

Table 27: Configuration setting: CollectPoi.QueryTimeoutInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config

Section name	<appSettings>
Setting	<add key ="CollectPoi.QueryTimeoutInMinutes" value="240"/>
Value	Default: 360 minutes
How to modify	Replace the value as required.

CollectPoi.QueryUpperBound

This key defines the maximum number of resources that will be held in memory as a result of the point of interest (POI) collection process. This key is used by the InfrastructureManagement internal service that handles general infrastructure management, including contacting the agent to retrieve POI information on governed resources.

Table 28: Configuration setting: CollectPoi.QueryUpperBound

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="CollectPoi.QueryUpperBound" value="1000000"/>
Value	Default: 1000000 resources
How to modify	Replace the value as required.

DfsDataSyncInterval

This key defines the preferred frequency (in minutes) at which the DFS synchronization occurs. This key is used by the DFSDataSync internal service that synchronizes managed DFS host information with the One Identity Manager database.

Table 29: Configuration setting: DfsDataSyncInterval

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="DFSDataSyncInterval" value="1440"/>
Value	Default: 1440 minutes (one day)
How to modify	Replace the value as required.

DirectAccessForServiceCredentials

This setting defines whether to enable the Data Governance service to access service account credentials directly from the database during agent deployment.

Table 30: Configuration setting: DirectAccessForServiceCredentials

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="DirectAccessForServiceCredentials" value="false"/>
Value	Valid values: • false: Data Governance service doesn't access service account credentials directly from the database during agent deployment • true: Data Governance service accesses service account credentials directly from the database during agent deployment
How to modify	Replace the value as required

DisablePerceivedOwnershipUpdate

This key can be used to disable the automatic perceived owner calculation for governed data. By disabling this calculation, custom perceived ownership calculations can be created by writing directly to the QAMPoIPerceievedOwner table.

Table 31: Configuration setting: DisablePerceivedOwnershipUpdate

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="DisablePerceivedOwnershipUpdate" value="false"/>
Value	Valid values: • false: Use the automatic perceived owner calculation for governed data (default). • true: Do not use the automatic perceived owner calculation for governed data.
How to modify	Replace the value as required.

EMCIsilonDFSLinksEnabled

This setting defines whether to enable the resolution of DFS links pointing to EMC Isilon CIFS device’s folder or not.

Table 32: Configuration setting: EMCIsilonDFSLinksEnabled

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="EMCIsilonDFSLinksEnabled" value="false"/>
Value	Valid values: • false: disable resolution of DFS links pointing to EMC Isilon CIFS Device’s folder • true: enable resolution of DFS links pointing to EMC Isilon CIFS Device’s folder
How to modify	Replace the value as required

EMCIsilonUseNetworkPathForAccessRequests

This setting defines whether to enable reading of access groups and their associated permissions directly from network share for resource access requests for DFS links.

Table 33: Configuration setting: EMCIsilonUseNetworkPathForAccessRequests

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key=" EMCIsilonUseNetworkPathForAccessRequests" value-e="false"/>
Value	Valid values: • false: disable reading access groups directly from network share for requests for DFS links • true: enable reading access groups directly from network share for requests for DFS links
How to modify	Replace the value as required

FolderSecurity.UseAdminPathsForShareFolders

This key controls how the server deals with the security that backs folders. There are two methods of getting and setting folder security when looking at a share path:

- You can do it with the \\comp\share path and a folder resource type
- You can get the folder path that backs the share, convert it to an admin share path and use a folder type.

Table 34: Configuration setting: FolderSecurity.UseAdminPathsForShareFolders

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="FolderSecurity.UseAdminPathsForShareFolders" value="0"/>
Value	Valid values: • 0: Use share permissions for the share (default) • 1: Use backing folder security for the share
How to modify	Replace the value as required.

ManagedHostDeleteBatchSize

This key is used by the methods that delete managed hosts and their associated resource and resource activity from the database. The data is deleted in batches of this size to avoid locking the database for the time it takes to delete it all, thus letting any other database activity to complete while this batch deletion is in progress.

Table 35: Configuration setting: ManagedHostDeleteBatchSize

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key "ManagedHostDeleteBatchSize" value="10000"/>
Value	Default: 10000
How to modify	Replace the value as required.

MessagingCacheFolder

This key defines the directory where the server's message store is to exist.

Table 36: Configuration setting: MessagingCacheFolder

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="MessagingCacheFolder" value=".\ServerMessagingCache"/>
Value	Directory for the server's message store.
How to modify	Replace value with the directory where the server's message store is to exist.

Metrics.CollectionIntervalInSeconds

This key defines the frequency (in seconds) at which metrics are collected. This key is used by the Metrics internal service which handles metric collection.

Table 37: Configuration setting: Metrics.CollectionIntervalInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="Metrics.CollectionIntervalInSeconds" value="60"/>
Value	Default: 60 seconds
How to modify	Replace the value as required.

MinimumSupportedModuleMigrationVersion

This key specifies the minimum supported version of the One Identity Manager database that this version of the Data Governance server can work with. If during installation, the new version is less than this value, the installation cannot occur. This key is used by the ManagementServer internal service which manages the core Data Governance service dependencies.

Table 38: Configuration setting: MinimumSupportedModuleMigrationVersion

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="MinimumSupportedModuleMigrationVersion" value="Null"/>
Value	Default: Null
How to modify	Replace the value as required.

OracleBulkImportBatchSize

> NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

Once data is 'governed', the Data Governance service periodically queries agents to retrieve detailed security information and store it in the central database. Use this configuration setting to configure the number of records to be imported at a time during a bulk import for an Oracle database.

> TIP: Configuring bulk inserts of governed security information may help improve performance.

Table 39: Configuration setting: OracleBulkImportBatchSize

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="OracleBulkImportBatchSize" value-"500"/>
Value	Default: 500
How to modify	Replace value as required.

PerceivedOwnershipActivityPeriod

This key defines the time period (in days) to look for past resource activity in order to determine perceived owners. This key is used by the ResourceActivityInternalServices task that updates the perceived owner and points of interest (POI) for governed resources.

Table 40: Configuration setting: PerceivedOwnershipActivityPeriod

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="PerceivedOwnershipActivityPeriod" value="30"/>
Value	Default: 30 days
How to modify	Replace the value as required.

PerceivedOwnershipByResourceActivity

This key specifies the primary source for calculating perceived ownership: Resource activity history or security information.

Table 41: Configuration setting: PerceivedOwnershipByResourceActivity

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="PerceivedOwnershipByResourceActivity" value="true"/>
Value	Valid values: • false: Use security information • true: Use resource activity history (default)
How to modify	Replace the value as required.

PerceivedOwnershipByResourceOwner

This key indicates whether the access control list owner within the target system should be considered as a perceived owner suggestion.

Table 42: Configuration setting: PerceivedOwnershipByResourceOwner

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config

Section name	<appSettings>
Setting	<add key ="PerceivedOwnershipByResourceOwner" value="false"/>
Value	Valid values: • false: Do not use access control list owner when calculating perceived owner (default). • true: Use access control list owner when calculating perceived owner.
How to modify	Replace the value as required.

PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes

This key defines the frequency (in minutes) at which the perceived owner information (resource activity) for resources is updated in the One Identity Manager dashboards. This key is used by the ResourceActivityInternalServices task that updates the perceived owner and points of interest (POI) for governed resources.

Table 43: Configuration setting: PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="PerceivedOwnershipCalcUpdatesRefreshIntervalMinutes" value="1440"/>
Value	Default: 1440 minutes (one day)
How to modify	Replace the value as required.

PerceivedOwnershipMaxReturnValue

This key defines the maximum number of perceived ownership suggestions to be returned when using the perceived owner calculation feature in Data Governance Edition.

Table 44: Configuration setting: PerceivedOwnershipMaxReturnValue

Configuration	%ProgramFiles%\One Identity\One Identity Manager Data

file	Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="PerceivedOwnershipMaxReturnValue" value="5"/>
Value	Default: 5 perceived owners
How to modify	Replace the value as required.

RemoteExecutor.WaitResultTimeout

This key defines how long (number of minutes) the Data Governance service should wait for results from the Remote Executor before timing out. The Remote Executor is used to deploy and uninstall agents.

Table 45: Configuration setting: RemoteExecutor.WaitResultTimeout

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="RemoteExecutor.WaitResultTimeout" value="5"/>
Value	Default: 5 minutes
How to modify	Replace the value as required.
Notes	During an agent uninstall, the Data Governance server sends a command to stop the agent service and then waits for the service to stop. If the agent service takes a long time to shut down, the uninstall fails. To solve this issue, the timeout value for the agent service stop can be increased using this configuration setting. In addition, if you modify the timeout value here, you should also update the StartStopServiceTimout registry key to the same value. For more information, see Agent start/stop timeout (StartStopServiceTimeout) on page 117.

RemoteHostForCloudOnlyImplementation

This configuration setting is used to provide an ability to specify and load a single Windows Server computer from Active Directory in the **Managed host** view in Manager application for deploying an agent for managing only cloud hosts. This key specifies the DNS host name of the Windows Server to be used for deploying the remote agent for cloud hosts. This key is only applicable for Windows Servers and not any other type of hosts.

Table 46: Configuration setting: RemoteHostForCloudOnlyImplementation

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="RemoteHostForCloudHostOnlyImplementation" value=""/>
Value	Valid values: • <empty>: All hosts are loaded in the **Managed host** view (default) • <DNS Host name of Windows Server>: The Windows Server whose DNS host name is specified is loaded in the **Managed host** view. Other Windows Servers aren’t loaded.
How to modify	Replace the value as required.

RestServicePort

This key sets the communication port for HTTP protocol and REST services. This port is used for communications with PowerShell and One Identity Manager clients and web server.

Table 47: Configuration setting: RestServiePort

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="RestServicePort" value="8723"/>
Value	Default: 8723
How to modify	Replace the value as required.

SelfService.AllowNonPublishedGroups

This setting defines whether groups that are not published to the IT Shop are to be displayed in the list of groups presented in the web portal for a self-service resource access request.

Table 48: Configuration setting: SelfService.AllowNonPublishedGroups

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="SelfService.AllowNonPublishedGroups" value="false"/>
Value	Valid values: • false: do not include non-published groups in list (default) • true: include non-published groups in list
How to modify	Replace the value as required.
Notes	This setting affects what groups are shown to the business owner.

SelfService.AllowUnsychronizedGroups

This setting defines whether groups that are not synchronized by One Identity Manager are to be displayed in the list of groups presented in the web portal for a self-service resource access request.

Table 49: Configuration setting: SelfService.AllowUnsychronizedGroups

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="SelfService.AllowUnsychronizedGroups" value="false"/>
Value	Valid values: • false: do not include unsychronized groups in list (default) • true: include unsychronized groups in list
How to modify	Replace the value as required.
Notes	This setting affects what groups are shown to the business owner.

SelfService.EnableSelfServiceRequest

This key is used to enable the self-service requests functionality.

Table 50: Configuration setting: SelfService.EnableSelfServiceRequest

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="SelfService.EnableSelfServiceRequest" value="true"/>
Value	Valid values: • false: disable self-service requests • true: enable self-service requests (default)
How to modify	Replace the value as required.

SelfService.IncludeSuitabilityTraceInfo

This key is used to turn on the suitability trace information property in the results of a self-service request. When enabled, additional trace logging is provided regarding the "best fit" group calculation.

Table 51: Configuration setting: SelfService.IncludeSuitabilityTraceInfo

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="SelfService.IncludeSuitabilityTraceInfo" value="false"/>
Value	Valid values: • false: do not include the additional trace information (default) • true: include the additional trace information. That is, the information is presented as a property of the returned objects.
How to modify	Replace the value as required.

SelfService.MarkSuitabilityVisible

This key indicates whether the suitability integer of the groups returned through a self-service request is visible in the user interface.

The suitability integer is the total after all of the calculators have been run and is used to rank the groups from which the business owner can select. The group with the highest total (suitability integer) is marked as the "best fit" group.

Table 52: Configuration setting: SelfService.MarkSuitabilityVisible

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="SelfService.MarkSuitabilityVisible" value="false"/>
Value	Valid values: • false: hide the suitability integer in the user interface (default) • true: show the suitability integer in the user interface
How to modify	Replace the value as required.

SelfService.MaximumMethodsCount

This key configures the maximum number of self-service groups to be returned for consideration. It is used by the AccessSelfService internal service that handles self-service requests initiated from the IT Shop.

Table 53: Configuration setting: SelfService.MaximumMetholdsCount

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="SelfService.MaximumMethodsCount" value="5"/>
Value	Default: 5
How to modify	Replace the value as required.

SelfService.SuitabilityThreshold

This key defines the lowest possible suitability index that can be returned by the self-service access request algorithm. Any group whose suitability drops below this threshold will be removed from the list of suitable groups. This key is used by the AccessSelfService internal service that handles self-service requests initiated through the IT Shop.

Table 54: Configuration setting: SelfService.SuitabilityThreshold

Configuration	%ProgramFiles%\One Identity\One Identity Manager Data

file	Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="SelfService.SuitabilityThreshold" value="100"/>
Value	Default: 100
How to modify	Replace the value as required.
Notes	This setting affects what groups are shown to the business owner.

SuggestedAgentCap

This key specifies the suggested maximum number of agent instances on a given computer.

Table 55: Configuration setting: SuggestedAgentCap

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key ="SuggestedAgentCap" value="20"/>
Value	Default: 20 agent instances
How to modify	Replace the value as required.
Notes	This is NOT a hard cap, just a suggestion.

SyncDomainPasswordInterval

This key defines the frequency (in seconds) at which the managed domain and security information cache is refreshed. This key is used by the GroupResolutionService internal service that maintains a cache of known managed domains and security information used for a variety of services.

Table 56: Configuration setting: SyncDomainPasswordInterval

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config

Section name	<appSettings>
Setting	<add key ="SyncDomainPasswordInterval" value="60"/>
Value	Default: 60 seconds
How to modify	Replace the value as required.

VerboseHostForTrusteeLogging

This key indicates whether to log the complete Alias table for a given query. This key is used for debugging purposes.

Table 57: Configuration setting: VerboseHostForTrusteeLogging

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server-\DataGovernanceEdition.Service.exe.config
Section name	<appSettings>
Setting	<add key="VerboseHostForTrusteeLogging" value="false"/>
Value	Valid values: • false: do not enable verbose logging (default) • true: enable verbose logging
How to modify	Change the value to "true" to enable verbose logging.
Notes	This setting should not be turned on for more time than is necessary to diagnose any issues encountered.

Activity weight multipliers

The activity weight multipliers in the Data Governance server configuration file affect the perceived owner calculations, which is based on the resource activity data collected by Data Governance agents. A weight is assigned to each different type of activity. The default calculation assumes that it is more likely that the data owner would create, edit, delete, and change security rather than just read the data, so a heavier weight has been assigned to these change operations. By default, the heaviest weight is given to change security and lightest weight to read.

For more information on business ownership and calculated perceived owners, see the *One Identity Manager Data Governance Edition User Guide.*

Table 58: Activity weight multipliers

Configuration setting	Description
<add key="Activity.ReadWeightMultiplier" valuee="100"/>	Weight assigned to read operations. By default, this is the lowest value.
<add key="Activity.WriteWeightMultiplier" valuee="150"/>	Weight assigned to write operations.
<add key="Activity.CreateWeightMultiplier" value="150"/>	Weight assigned to create operations.
<add key="Activity.DeleteWeightMultiplier" value="150"/>	Weight assigned to delete operations.
<add key="Activity.RenameWeightMultiplier" value="125"/>	Weight assigned to rename operations.
<add key="Activity.SecurityChangeWeightMultiplier" value="200"/>	Weight assigned to security changes. By default, this is the highest value.

To configure the perceived owner calculation

1. Browse to and open the DataGovernanceEdition.Service.exe.config file.
2. In the configuration file, locate the Application settings (<appSettings>) section
3. Locate and alter the value assigned to the required key.
4. Save your changes.
5. Restart the Data Governance service after making changes to these settings and saving the file.

Self-service suitability calculation multipliers

The "best fit" group is determined through a series of calculators that work on various criteria. Each calculator returns a value in the range of -2 to +2:

- Very Bad (-2)
- Bad (-1)
- Neutral (0)
- Good (+1)
- Very Good (+2)

These calculators cannot be changed, but you can modify the positive and negative multipliers by changing the default values defined in the DataGovernanceEdition.Service.exe.config file. The following set of multipliers are used by the self-service calculation system to modify the relative weights of the various suitability calculators.

Keep in mind that the multiplier values are only relative to one another. If you doubled all the multipliers, there would be no change in the resulting set of groups returned to the user. If you want your desired criteria to be considered more importance, set the multipliers on those calculators to be higher relative to the rest.

NTFS group membership calculation multipliers

Configuration settings:

<add key="SelfService.AccessInheritanceSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.AccessInheritanceSuitabilityProcessor.NegativeMultiplier" value="100"/>

> Checks access inheritance: Groups whose rights to the targeted resource are explicit are favorable. Groups that have been delegated access to the targeted resource through inherited permissions are considered less favorable.
>
> - If the permissions have been inherited from some resource higher in the hierarchy, then the requester may be given access to more resources than they've actually requested. (Bad)
> - If nothing is gained through inherited access, don't change the suitability. (Neutral)
> - If the explicitly held rights are a better match than neutral and there are no inherited rights, then that's good (Good)

<add key="SelfService.AccessSuitabilityProcessor.PositiveMultiplier" value="200"/>

<add key="SelfService.AccessSuitabilityProcessor.NegativeMultiplier" value="500"/>

> Checks access rights:
>
> - It is optimal if the access held by the group is exactly what the request requires. (Very good)
> - If the group has slightly more access than is required, it may be suggested but considered less favorable. (Good).
> - It is detrimental if the group has "dangerous" rights, such as Full Control, Take Ownership, or Change Permissions. (Very bad)
> - If the group doesn't have sufficient access to meet the request, it is marked as ineligible for selection. (ineligible).

<add key="SelfService.DomainLocalMembershipSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.DomainLocalMembershipSuitabilityProcessor.NegativeMultiplier" value="200"/>

> Checks Domain Local group membership:

- If a group contains any Global or Universal groups, then it's likely being used as a resource group. This means that the group should be less desirable for usage as an access provisioning group. (Bad)
- If a group does not contain any Global or Universal groups, then it is most likely used for direct access provisioning and not as a container group. (Very good)

<add key="SelfService.DomainLocalMembershipSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.DomainLocalMembershipSuitabilityProcessor.NegativeMultiplier" value="200"/>

Checks group membership rules:

- Global groups that exist in the same domain as the identity are favorable.
- If the group is Universal, the identity must exist in the same forest as the group.

> NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.GroupTypeSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.GroupTypeSuitabilityProcessor.NegativeMultiplier" value="200"/>

Checks group type: Based on Microsoft best practices, groups are favored in the following order:

- If the group is a Global group, it is marked as very good.
- If the group is a Universal group, it is marked as good.
- If the group is a Domain Local group, it is marked as bad.
- Domain built-in groups and non-security groups are never considered suitable selections and are marked as ineligible.

<add key="SelfService.OriginInformationSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.OriginInformationSuitabilityProcessor.NegativeMultiplier" value="100"/>

Check origin domain:

- Groups in the same domain as the requesting identity are considered favorable. (Very good)
- Groups from the resource's forest are considered less favorable. (Good)
- Groups from forests outside of the forest of the requesting identity are considered even less favorable. (Bad)

<add key="SelfService.ResourceDistanceSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.ResourceDistanceSuitabilityProcessor.NegativeMultiplier" value="100"/>

Checks distance from resource: The closer the group is to the resource, the better. The further away the groups gets from the ACL, the wore the score.

- Groups directly in the resources access control list are considered favorable.
- A group that is nested one or more steps away from the access control list is considered less favorable.

> NOTE: This calculator never marks a group as very bad.

SharePoint group access calculation multipliers

Configuration settings:

<add key="SelfService.BestFitPermissionLevelSuitabilityProcessor.PositiveMultiplier" value="300"/>

<add key="SelfService.BestFitPermissionLevelSuitabilityProcessor.NegativeMultiplier" value="100"/>

Choose a group assigned a permission level that best fits the requested access. Not enough rights makes the group Ineligible. Granting any modification permissions when only Contribute permissions are requested makes the group ineligible.

<add key="SelfService.DelegationGrantingPermissionLevelSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.DelegationGrantingPermissionLevelSuitabilityProcessor.NegativeMultiplier" value="100"/>

Groups that contain permission levels that grant a user not only the requested rights, but also give the ability to delegate permissions to others will be marked as ineligible.

<add key="SelfService.FarmAdminAvoidSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.FarmAdminAvoidSuitabilityProcessor.NegativeMultiplier" value="100"/>

Avoid groups that grant farm administrative rights. Farm Admin groups are marked as ineligible, otherwise the group is marked as neutral.

> NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.JoinOptionsSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.JoinOptionsSuitabilityProcessor.NegativeMultiplier" value="100"/>

Checks a group's access properties:

- If the group is not a SharePoint group, it is marked as neutral.
- If the auto-accept members flag is set, the group is assumed to be extremely safe and it is marked as very good.
- If a workflow exists for granting access, or current members of the group are able to add others, the group is marked as good.
- If the property that specifies only group members may view the membership is set, the group is assumed to be fairly locked down; therefore, the group is marked as bad.

<add key="SelfService.PermissionsAgreeSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.PermissionsAgreeSuitabilityProcessor.NegativeMultiplier" value="100"/>

Many Windows groups that may be viable through Windows Domain Trusts do not always work in granting SharePoint access because of limitations in SharePoint security checking. This calculator checks to see if SharePoint itself considers the group valid for the requested access. If the effective permissions meet the requirements of the requested permissions, that is very good. Otherwise, it is marked as neutral.

> NOTE: Since this calculator only marks a group as very good or neutral, changing a multiplier will not change the results.

<add key="SelfService.NestingSuitabilityProcessor.PositiveMultiplier" value="200"/>

<add key="SelfService.NestingSuitabilityProcessor.NegativeMultiplier" value="100"/>

If the target group is an Active Directory group that is also a member of a SharePoint group, it is marked as very good. Otherwise, it is marked as neutral.

> NOTE: Since this calculator only marks a group as very good or neutral, changing a multiplier will not change the results.

<add key="SelfService.PreferActiveDirectoryGroupTypeSuitabilityProcessor.PositiveMultiplier" value="50"/>

<add key="SelfService.PreferActiveDirectoryGroupTypeSuitabilityProcessor.NegativeMultiplier" value="100"/>

Checks the type of group:

- If the group is a SharePoint group, it is marked as neutral.
- If the group is a security-enabled Active Directory group, it is marked as ineligible.
- If the group is a global Active Directory group, it is marked as very good.
- If the group is a universal Active Directory group, it is marked as good.
- If the group is a built-in domain group, it is marked as ineligible.
- If the group is a local domain group, it is marked as bad.

NOTE: The default values when none of these are satisfied mark the group as ineligible.

<add key="SelfService.PreferSharePointGroupTypeSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.PreferSharePointGroupTypeSuitabilityProcessor.NegativeMultiplier" value="100"/>

Some organizations prefer to use groups that are SharePoint groups because they enhance SharePoint features and delegation within SharePoint itself, as well as allowing self service. This is a trade-off between SharePoint features vs. Active Directory group power in the enterprise. The use of Active Directory groups vs. SharePoint groups as a best practice is a debated topic.

If a group is a SharePoint group, mark it as very good, otherwise mark it as neutral. To avoid SharePoint groups, flip the positive “weight” to a negative number.

<add key="SelfService.SiteCollectionAvoidAdminSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.SiteCollectionAvoidAdminSuitabilityProcessor.NegativeMultiplier" value="100"/>

Avoid groups that grant Site Collection Administrative rights. These groups are marked as ineligible. Otherwise, the group is marked as neutral.

NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.WebAppPolicyAvoidActAsSystemSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.WebAppPolicyAvoidActAsSystemSuitabilityProcessor.NegativeMultiplier" value="100"/>

Avoid groups that would cause the user to gain the Act As System right. These groups are marked as ineligible. Otherwise, the group is marked as neutral.

NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.WebAppPolicyAvoidSiteCollectionRightsSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.WebAppPolicyAvoidSiteCollectionRightsSuitabilityProcessor.NegativeMultiplier" value="100"/>

Avoid groups that Web Application policies grant Site Collection Administrative rights to. These groups are marked as ineligible. Otherwise, the group is marked as neutral.

NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.WebAppPolicyDenySuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.WebAppPolicyDenySuitabilityProcessor.NegativeMultiplier" value="100"/>

Some Farms may have policies denying most users from ever getting permissions that are too high.

- Any rights denied outside the requested permissions are considered neutral.
- A policy can make the group ineligible if it denies rights being requested.

> NOTE: Since this calculator only marks a group with ineligible or neutral, changing a multiplier will not change the results.

<add key="SelfService.WebAppPolicyGrantSuitabilityProcessor.PositiveMultiplier" value="100"/>

<add key="SelfService.WebAppPolicyGrantSuitabilityProcessor.NegativeMultiplier" value="100"/>

Avoid groups that get rights granted via a Web Application policy (in any zone). The more rights granted, the worse it is. These policies are usually used to grant service accounts, like the Search Service accounts rights, and are not generally good ways to obtain access to resources.

- If the group has MORE than the following permissions, then it is marked as ineligible:
  - LIST PERMISSIONS: ViewItems, ViewApplicationPages, OpenItems, ViewVersions, CreateAlerts, ViewApplicationPages
  - SITE PERMISSIONS: ViewPages, Open, ViewPages, BrowseUserInformation, UseRemoteInterfaces, UseClientIntegrationFeatures, Open, UseSelfServiceSiteCreation, EditPersonalUserInformation, ApplyThemesAndBorders, ApplyStyleSheets
  - PERSONAL PERMISSIONS: ManagePersonalViews, AddRemovePersonalWebParts, UpdatePersonalWebParts
- If the group has MORE than the following permissions, then it is marked as very bad:
  - LIST PERMISSIONS: ViewItems, ViewApplicationPages, OpenItems, ViewVersions, CreateAlerts, ViewApplicationPages
  - SITE PERMISSIONS: ViewPages, Open, ViewPages, BrowseUserInformation, UseRemoteInterfaces, UseClientIntegrationFeatures, Open, UseSelfServiceSiteCreation, EditPersonalUserInformation
  - PERSONAL PERMISSIONS: ManagePersonalViews, AddRemovePersonalWebParts, UpdatePersonalWebParts
- If the group has the EXACTLY the following permissions, then it is marked as bad:

- LIST PERMISSIONS: ViewItems, ViewApplicationPages, OpenItems, ViewVersions, CreateAlerts, ViewApplicationPages
- SITE PERMISSIONS: ViewPages, Open, ViewPages, BrowseUserInformation, UseRemoteInterfaces, UseClientIntegrationFeatures, Open, UseSelfServiceSiteCreation, EditPersonalUserInformation
- PERSONAL PERMISSIONS: ManagePersonalViews, AddRemovePersonalWebParts, UpdatePersonalWebParts

Data Governance agent configuration file settings

The following Data Governance agent configuration file settings can be configured in the DataGovernance.Agent.exe.config file in the Agent Services directory in the agent's installation directory: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services.

Table 59: Communication settings

Configuration setting	Description
baseActivePort	Sets the default listening port.
cloudScanThreadMax	Sets the maximum number of concurrent scan threads to be used when scanning a cloud managed host.
overrideServerUri	Indicates that the agent is to connect to a specific Uri and not use the results from an Active Directory service connection point search.
shimCloseTimeoutInMinutes	Dictates the interval of time provided for a connection to the Shim to close before the transport raises an exception. NOTE: SharePointShim is used when monitoring a SharePoint 2010 host.
shimOpenTimeoutInMinutes	Dictates the interval of time provided for a connection to open to the Shim before the transport raises an exception. NOTE: SharePointShim is used when monitoring a SharePoint 2010 host.
shimReceiveTimeoutInMinutes	Dictates the interval of time that a connection can remain inactive, during which time no application messages are received from the Shim before it is dropped.

Configuration setting	Description
	NOTE: SharePointShim is used when monitoring a SharePoint 2010 host.
shimSendTimeoutInMinutes	When writing to the Shim, this setting dictates the interval of time provided for a write operation to complete before the transport raises an exception. NOTE: SharePointShim is used when monitoring a SharePoint 2010 host.

Table 60: Windows computer settings

Configuration setting	Description
indexingEnabled (localGroup scanning)	Determines whether local group scanning is enabled.
indexingEnabled (local user rights scanning)	Determines whether local user rights scanning is enabled.
indexingEnabled (share scanning)	Determines whether share scanning is enabled.
localGroupResolutionInSeconds	Sets the number of seconds between scans of local groups.
windowsComputerResourceResolutionInSeconds	Sets the number of seconds between full scans of the various resources within the Windows Computer resource namespace.

Table 61: Service identity indexer settings

Configuration setting	Description
indexingEnabled (service identities scanning)	Determines whether service identities scanning is enabled.
serviceIdentityIndexingResolutionInSeconds	Sets the number of seconds between scans of service identities.

Table 62: Security data store service setting

Configuration setting	Description
keepQueryDocuments	Diagnostic setting used to debug or diagnose issues with agent queries. This setting should only be enabled for diagnostic purposes as it will save *raq files to the agent instance folder and not delete them. These can eventually take up a large amount of disk space.

Table 63: Resource usage settings

Configuration setting	Description
numberOfSharepointScanThreads	Defines the number of threads to be used when the agent is scanning the SharePoint object hierarchy in the farm.
usageFlushIntervalInSeconds	Sets the frequency (in seconds) at which auditing information being held in memory is flushed to disk.

Table 64: NetApp configuration setting

Configuration setting	Description
OverrideFPolicyName	Overrides the name of the policy the FPolicy change watcher connects to.

Table 65: Cloud configuration setting

Configuration setting	Description
cloudGroupResolutionInSeconds	Sets the number of seconds between scans of cloud groups.

baseActivePort

Use this setting to change the default listening port.

Table 66: Agent configuration setting: baseActivePort

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="baseActivePort" type="dword">
Value	Default: 18530
How to modify	1. Stop the agent service. 2. Change the baseActivePort, replacing the value as required. 3. Start the agent service.
Notes	The agent starts with this port and if it can not get this port, increases it by one until it can open the listening port.

cloudGroupResolutionInSeconds

Use this configuration setting to change the number of seconds between scans of cloud groups.

Table 67: Agent configuration setting: cloudGroupResolutionInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="localGroup">
Setting	<Setting name ="cloudGroupResolutionInSeconds" type= "dword">
Value	Default: 86400 seconds (which is once a day)
How to modify	Replace value as required.

cloudScanThreadMax

Use this configuration setting to set the maximum number of concurrent scan threads to be used when scanning a cloud managed host.

Table 68: Agent configuration setting: cloudScanThreadMax

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="cloudScanThreadMax" type="dword">
Value	Default: 4
How to modify	Replace value as required.
Notes	If the scanner is throttling, reduce to 1 or 2 threads. A reasonable maximum is 16.

indexingEnabled (localGroup scanning)

This setting determines whether localGroup scanning is enabled. By default, the server sets this to 1 indicating that localGroup scanning is enabled.

> NOTE: This just controls the synchronization. The local groups are always scanned because the data is required for access calculations.

Table 69: Agent configuration setting: indexingEnabled

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="localGroup">
Setting	<Setting name ="indexingEnabled" type="dword">
Value	Valid values: • 0: disable localGroup scanning • 1: enable localGroup scanning (default)
How to modify	Replace value as required.

indexingEnabled (local user rights scanning)

This setting determines whether local user rights scanning is enabled. By default, the server sets this to 1 indicating that local user rights scanning is enabled.

Table 70: Agent configuration setting: indexingEnabled

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="localUserRights">
Setting	<Setting name ="indexingEnabled" type="dword">
Value	Valid values: • 0: disable localUserRights scanning • 1: enable localUserRights scanning (default)
How to modify	Replace value as required.

indexingEnabled (service identities scanning)

This setting determines whether service identities scanning is enabled. By default, the server sets this to 1 indicating that service identities scanning is enabled.

Table 71: Agent configuration setting: indexingEnabled

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="securityIdentityIndexer">
Setting	<Setting name ="indexingEnabled" type="dword">
Value	Valid values: • 0: disable services identity scanning • 1: enable services identity scanning (default)
How to modify	Replace value as required.

indexingEnabled (share scanning)

This setting determines whether share scanning is enabled. By default, the server sets this to 1 indicating that share scanning is enabled.

Table 72: Agent configuration setting: indexingEnabled

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="share">
Setting	<Setting name ="indexingEnabled" type="dword">
Value	Valid values: • 0: disable share scanning • 1: enable share scanning (default)
How to modify	Replace value as required.

keepQueryDocuments

This a diagnostic setting and is useful when debugging or diagnosing issues with agent queries. When this configuration setting is enabled, agents record the responses they send to the Data Governance server for ALL queries.

Table 73: Agent configuration setting: keepQueryDocuments

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="SecurityDataStore">
Setting	<Setting name ="keepQueryDocuments" type="dword">
Value	Valid values: • 0: disabled (default) • 1: enabled
How to modify	Replace value as required.

Notes	The response documents generated by this setting can be very large and will create a significant amount of data on the disk. This setting should not be turned on for more time than is necessary to diagnose any issues encountered.

localGroupResolutionInSeconds

Use this configuration setting to change the number of seconds between scans of local groups.

Table 74: Agent configuration setting: localGroupResolutionInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="localGroup">
Setting	<Setting name ="localGroupResolutionInSeconds" type="dword">
Value	Default: 86400 seconds (which is once a day)
How to modify	Replace value as required.

numberOfSharepointScanThreads

Use this configuration setting to define the number of threads used when the agent is scanning the SharePoint object hierarchy in the farm.

Table 75: Agent configuration setting: numberOfSharepointScanThreads

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="ResourceUsage">
Setting	<Setting name ="numberOfSharepointScanThreads" type="dword">
Value	Default: 30 threads
How to modify	Replace value as required.

OverrideFPolicyName

When working with NetApp filer devices, it may be useful to use a shorthand name, especially when working with simulator devices. You can use this configuration setting to override the name of the policy the FPolicy change watcher connects to. You can also use this configuration setting to specify the name of a manually created FPolicy.

Table 76: Configuration setting: OverrideFPolicyName

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<"Agent"> <"Services"> <"ChangeMonitoring">
Setting	<Setting name="OverrideFPolicyName" type="dword">
Value	FPolicy name the agent is to register with.
How to modify	1. Stop the agent service. 2. Set the overrideFPolicyName. 3. Start the agent service.
Notes	This configuration setting only applies to NetApp 7-Mode devices with CIFS file system protocol enabled. If you have an FPolicy defined for the Data Governance agent outside of the One Identity Manager framework, use this configuration setting to enter the exact FPolicy name. When an FPolicy name is set using this configuration setting, the Data Governance server will not create an FPolicy and the Data Governance agent will update the FPolicy name provided when communicating with the NetApp device. If the FPolicy name is not defined using this configuration setting, the Data Governance server creates an FPolicy with a name that matches the agent instance folder name located on the server where the agent runs. An example of an FPolicy name automatically created by the Data Governance server is "DGE_TEST_myNetApp_myDomain_local".

overrideServerUri

Set this configuration setting if you want the agent to connect to a specific Uri and not use the results from an Active Directory service connection point (SCP) search.

Table 77: Agent configuration setting: overrideServerUri

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="overrideServerUri" type="dword">
Value	The Uri to be used. Use the net.tcp format, for example: net.tcp://myDGEServerHost.myDomain.local:8722/Broadway/IndexServerAgent.svc
How to modify	1. Stop the agent service. 2. Set the overrideServerUri. 3. Start the agent service.
Notes	Use this setting to force a connection when SCP objects cannot be created in the domain where an agent resides or you do not want to use Active Directory queries to find the Data Governance server SCP in the forest and connect to the Data Governance Edition URI configured in the SCP. Set the value to the same URI listed in the Data Governance service app.config (DataGovernanceEdition.Service.exe.config) under one o f the baseAddresses for <service name="QAM.Server.ServiceHosts.IndexServerHost">.

serviceIdentityIndexingResolutionInSeconds

Use this configuration setting to change the number of seconds between scans of service identities.

Table 78: Agent configuration setting: serviceIdentityIndexingResolutionInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="serviceIdentityIndexer">

Setting	<Setting name ="serviceIdentityIndexingResolutionInSeconds" type="dword">
Value	Default: 120 seconds
How to modify	Replace value as required.

shimCloseTimeoutInMinutes

The SharePointShim is used whenever you are monitoring a SharePoint 2010 host. This setting dictates the interval of time provided for a connection to the Shim to close before the transport raises an exception.

> NOTE: This is a global setting and applies to all SharePointShim processes used for a multi-agent SharePoint 2010 managed host.

Table 79: Agent configuration setting: shimCloseTimeoutInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernanceEdition.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="shimCloseTimoutInMinutes" type="dword">
Value	Default: 1 minute Maximum value is 7 days.
How to modify	1. Stop the agent service. 2. Change the shimCloseTimeoutInMinutes, replacing the value as required. 3. Start the agent service.
Notes	This setting is not configurable through the UI.

shimOpenTimeoutInMinutes

The SharePointShim is used whenever you are monitoring a SharePoint 2010 host. This setting dictates the interval of time provided for a connection to open to the Shim before the transport raises an exception.

> NOTE: This is a global setting and applies to all SharePointShim processes used for a multi-agent SharePoint 2010 managed host.

Table 80: Agent configuration setting: shimOpenTimeoutInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernanceEdition.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="shimOpenTimoutInMinutes" type="dword">
Value	Default: 1 minute Maximum value is 7 days.
How to modify	1. Stop the agent service. 2. Change the shimOpenTimeoutInMinutes, replacing the value as required. 3. Start the agent service.
Notes	This setting is not configurable through the UI.

shimReceiveTimeoutInMinutes

The SharePointShim is used whenever you are monitoring a SharePoint 2010 host. This setting dictates the interval of time that a connection can remain inactive, where no application messages are received from the Shim before it is dropped.

> NOTE: This is a global setting and applies to all SharePointShim processes used for a multi-agent SharePoint 2010 managed host.

Table 81: Agent configuration setting: shimReceiveTimeoutInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernanceEdition.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="shimReceiveTimoutInMinutes" type="dword">
Value	Default: 10 minutes Maximum value is 7 days.
How to modify	1. Stop the agent service.

	2. Change the shimReceiveTimeoutInMinutes, replacing the value as required. 3. Start the agent service.
Notes	This setting is not configurable through the UI.

shimSendTimeoutInMinutes

The SharePointShim is used whenever you are monitoring a SharePoint 2010 host. When writing to the Shim, this setting dictates the interval of time provided for a write operation to complete before the transport raises an exception.

> NOTE: This is a global setting and applies to all SharePointShim processes used for a multi-agent SharePoint 2010 managed host.

Table 82: Agent configuration setting: shimSendTimeoutInMinutes

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernanceEdition.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="communication">
Setting	<Setting name ="shimSendTimoutInMinutes" type="dword">
Value	Default: 1 minutes Maximum value is 7 days.
How to modify	1. Stop the agent service. 2. Change the shimSendTimeoutInMinutes, replacing the value as required. 3. Start the agent service.
Notes	This setting is not configurable through the UI.

usageFlushIntervalInSeconds

Use this configuration setting to define the frequency (in seconds) at which auditing information being held in memory is flushed to disk.

Table 83: Agent configuration setting: usageFlushIntervalInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="ResourceUsage">
Setting	<Setting name ="usageFlushIntervalInSeconds" type="dword">
Value	Default: 10 seconds
How to modify	Replace value as required.
Notes	The shorter the interval, the smaller the window in which potential data loss can occur. Each flush operations causes disk access, which may lead to undue disk space usage it if the value is set too low.

windowsComputerResourceResolutionInSeconds

Use this setting to change the number of seconds between full scans of the various resources within the Windows Computer resource namespace.

Table 84: Agent configuration setting: windowsComputerResourceResolutionInSeconds

Configuration file	%ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services\DataGovernance.Agent.exe.config
Section name	<Section name="Agent"> <Section name="Services"> <Section name="windowsComputer">
Setting	<Setting name ="windowsComputerResourceResolutionInSeconds" type="dword">
Value	Default: 60 seconds
How to modify	Replace value as required.

9

Configurable registry settings

There are registry settings that can be configured for the Data Governance service.

NOTE: Legacy Data Governance agent registry settings are no longer available. Use the agent's configuration file to modify agent configurations that are not available in the Manager client. For more information, see Data Governance agent configuration file settings on page 91.

IMPORTANT: One Identity does not provide support for problems that arise from improper modification of the registry. The Windows registry contains information critical to your computer and applications. Make sure you back up the registry before modifying it. For more information on the Windows Registry Editor and how to back up and restore it, refer to Microsoft Article ID 256986: Windows registry information for advanced users, on the Microsoft support site.

Data Governance service registry settings

The following Data Governance service settings can be altered or created in the registry to modify the default behavior.

NOTE: After modifying a registry key, restart the Data Governance service and Manager to apply the changes.

Table 85: Registry key settings: HKEY_CURRENT_USER

Registry key setting	Description
Agent query timeout (AsyncQueryTimeoutInMinutes)	Specifies the maximum amount of time (in minutes) an agent query can run before it times out.
Resource access data points (MaxDataPoints)	Specifies the maximum number of data points to be included in a Resource Access report.

Registry key setting	Description
View deviations data points (MaxDataPoints)	Specifies the maximum number of data points to be included when viewing deviations.
Data governance overview results (MaxResults)	Specifies the maximum number of records to be returned and displayed on the Data governance overview.
WCF timeouts (wcfTimeoutInMinutes)	Specifies the maximum amount of time it should take a WCF command to complete before it times out.
Reporting timeout (WcfTimeoutReportingInMinutes)	Specifies the maximum amount of time it should take to generate a report before it times out.

Table 86: Registry key settings: HKEY_LOCAL_MACHINE

Registry key setting	Description
Write default classification level data to database (ClassificationLevelDefaultData)	Indicates whether the default classification level data is to be written to the One Identity Manager database.
Default identity SID (DefaultEmployeeSid)	Specifies the SID of the default identity used by the Data Governance topology harvest process.
Explicit exclusion of groups (ExclusionByDN)	Indicates whether to exclude groups from self-service group selection.
Filter accounts from Manage Access view (FilterNoisyAccounts)	Determines whether to filter out noisy accounts (that is, built-in accounts (Administrators and Users)) from the **Manage Access** view.
Global agent installation location (GlobalAgentInstallLocation)	Specifies the default installation location for deploying Data Governance agents.
Manual FPolicy creation (ManualFPolicyCreation)	Determines whether to manually create the FPolicy for a NetApp filer.
Resource Activity database connection string (QAMAuditActivityDBConnectionString)	Specifies the connection string to the Data Governance Resource Activity database.
Deployment name (QAMDeploymentId)	Specifies the deployment name assigned to the Data Governance Edition deployment.
Oracle deployment: Resource Activity database (QDGDBPlatformOracle)	Indicates whether you are using a SQL Server or Oracle database for the Resource Activity database.

Registry key setting	Description
	NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.
One Identity Manager database connection string (Q1IMDBConnectionString)	Specifies the connection string to the One Identity Manager database.
Oracle deployment: One Identity Manager database (Q1IMDBPlatformOracle)	Indicates whether you are using a SQL Server or Oracle database for the One Identity Manager database. NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.
Write default managed resource data to database (ResourceTemplateDefaultData)	Indicates whether the default managed resource data is written to the One Identity Manager database.
Agent start/stop timeout (StartStopServiceTimeout)	Sets the amount of time to wait for the agent service to start or stop before it times out.

Agent query timeout (AsyncQueryTimeoutInMinutes)

Create the following registry key on the client computer where the Manager is installed to specify the maximum amount of time (in minutes) an agent query can run before it times out.

Table 87: Registry setting: AsyncQueryTimeoutInMinutes

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client\Controls NOTE: The Controls subkey does not exist by default and will need to be created.
Value name	AsyncQueryTimeoutInMinutes
Value type	REG_DWORD
Value	Maximum amount of time, in minutes, before an agent query times out. Default: 20 minutes

Write default classification level data to database (ClassificationLevelDefaultData)

This key indicates whether the default classification levels defined in Data Governance Edition are written to the One Identity Manager database.

> NOTE: This registry value is checked on Data Governance service startup and if not present or if its value is set to 0, Data Governance Edition writes the default classification values into the One Identity Manager database and sets the registry value. When this value is set to 1, this indicates that the default classification level data is already stored in One Identity Manager database and should not be overwritten on service startup.

Table 88: Registry setting: ClassificationLevelDefaultData

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	ClassificationLevelDefaultData
Value type	REG_DWORD
Value	Valid values: • 0: Write the default classification level data into the One Identity Manager database. • 1: Default classification level data is already stored in the One Identity Manager database: do not overwrite on Data Governance service startup.
Notes	If you delete the default classification levels in your Data Governance Edition deployment and replace them with new classification levels, you must move or set this registry key if you move the Data Governance service to another machine. When you move the Data Governance service to another machine, before starting the Data Governance service ensure that this registry key is set (value is set to 1); otherwise, the Data Governance service will reload any previously deleted default database data that was inserted when the Data Governance service was initially started (on the first machine). If you modify the default classification levels in your Data Governance Edition deployment, the classification level data is retained if you move the Data Governance service to another machine.

Default identity SID (DefaultEmployeeSid)

This registry key specifies the security identifier (SID) of the default identity used by the Data Governance topology harvest process. This setting is used by the ManagementServer internal service that manages the core Data Governance service dependencies.

Table 89: Registry setting: DefaultEmployeeSid

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	DefaultEmployeeSid
Value type	REG_SZ
Value	SID of the user used by the Data Governance topology harvest process.
Note	This key is present if you used the Data Governance Configuration wizard to install the Data Governance service.

Explicit exclusion of groups (ExclusionByDN)

On the Data Governance server, configure the following registry key to exclude groups from self-service group selection.

> NOTE: You may want to mark certain groups as being ineligible for self-service requests, especially when Data Governance Edition is configured to allow for non-published groups to be presented. In this case, it is possible to mark either specific groups, or all groups within a particular Active Directory container as being ineligible for access requests.

Table 90: Registry setting: ExclusionByDN

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server\DeploymentData\SelfService NOTE: If the DeploymentData and SelfService subkeys do not exist, create them.
Value name	ExclusionByDN
Value type	REG_SZ
Value	Create string values whose names match the distinguished name of the

	groups that are to be excluded. To exclude an entire container of groups, specify the distinguished name of the container, with an asterisk ("*") prefix. For example, to exclude all groups in the Users container of example.com. use the following syntax: "*CN=Users,DC=example,DC=com".

Filter accounts from Manage Access view (FilterNoisyAccounts)

Create the following registry key on the client computer where the Manager is installed to indicate whether noisy accounts (that is, accounts with indirect access granted through the BUILTIN\Administrators or BUILTIN\Users accounts), are to be filtered from the **Manage Access** view.

Table 91: Registry setting: FilterNoisyAccounts

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Client
Value name	FilterNoisyAccounts
Value type	DWORD
Value	Valid values: • 0: do not filter out noisy accounts • 1: filter out noisy accounts (default)

Global agent installation location (GlobalAgentInstallLocation)

By default, the agent will be installed in %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Agent Services. To change this default location, create the following new string value in the registry on the Data Governance server.

Table 92: Registry setting: GlobalAgentInstallLocation

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	GlobalAgentInstallLocation

Value type	REG_SZ
Value	Agent installation location
Note	All agents attempt to deploy the folder specified in this registry setting. If, when you deploy an individual agent, you select an alternate installation location on the **Managed Hosts Settings** dialog, the location specified takes precedence over the default location specified in the registry.

Manual FPolicy creation (ManualFPolicyCreation)

On the Data Governance server, create the following registry key to manually create the FPolicy for a NetApp filer. Creating this registry key prevents the automatic creation of FPolicy on the specified NetApp filer.

Table 93: Registry setting: ManualFPolicyCreation

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	ManualFPolicyCreation
Value type	REG_SZ
Value	Fully qualified domain name of the NetApp filer.
Note	Ensure that the registry key has been created on the server before deploying the agent. In addition, you must also create a new configuration file, DefaultOntapSetting.qamel, in the Data Governance server folder: %ProgramFiles%\One Identity\One Identity Manager Data Governance Edition\Server.

Resource access data points (MaxDataPoints)

Create the following registry key on the client computer where the Manager is installed to specify the maximum number of data points to be included in a Resource Access report.

Table 94: Registry setting: MaxDataPoints

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client\Reporting\ResourceAccessReport NOTE: The Reporting and ResourceAccessReport subkeys do not exist by default and will need to be created.
Value name	MaxDataPoints
Value type	DWORD
Value	Maximum number of data points. Default: 10000

View deviations data points (MaxDataPoints)

Create the following registry key on the client computer where the Manager is installed to specify the maximum number of data points to be included when viewing deviations.

Table 95: Registry setting: MaxDataPoints

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client\Controls\ViewDeviations NOTE: The Controls and ViewDeviations subkeys do not exist by default and will need to be created.
Value name	MaxDataPoints
Value type	DWORD
Value	Maximum number of data points. Default: 10000

Data governance overview results (MaxResults)

Create the following registry key on the client computer where the Manager is installed to specify the maximum number of records to be returned and displayed on the Data governance overview.

Table 96: Registry setting: MaxResults

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client\DataUnderGovernanceView NOTE: The DataUnderGovernanceView subkey does not exist by default and will need to be created.
Value name	MaxResults
Value type	DWORD
Value	Maximum number of records to be returned and displayed. Default: 5000 records

Resource Activity database connection string (QAMAuditActivityDBConnectionString)

This registry setting specifies the connection string to the Data Governance Resource Activity database. This setting is used by the ManagementServer internal service that manages the core Data Governance service dependencies.

Table 97: Registry setting: QAMAuditActivityDBConnectionString

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	QAMAuditActivityDBConnectionString
Value type	REG_SZ
Value	Connection string assigned to the Resource Activity database.

Deployment name (QAMDeploymentId)

This registry key specifies the deployment name assigned to the Data Governance Edition deployment.

Table 98: Registry setting: QAMDeploymentId

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	QAMDeploymentId
Value type	REG_SZ
Value	Deployment name assigned to the Data Governance Edition deployment.
Note	In a new Data Governance Edition deployment, the default deployment name is DEFAULT.

Oracle deployment: Resource Activity database (QDGDBPlatformOracle)

> NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

This registry key defines the database system, SQL Server or Oracle Database, being used for the Resource Activity database. This setting is used by the ManagementServer internal service that manages the core Data Governance service dependencies.

> NOTE: If you are using an Oracle database for the Resource Activity database, create this registry key on the computer where the Data Governance service is to be installed. Once this key and the Q1IMDBPlatformOracle key are created and set to 1, use the Data Governance Configuration wizard to deploy the Data Governance service and create the Resource Activity database.

Table 99: Registry setting: QDGDBPlatformOracle

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	QDGDBPlatformOracle
Value type	REG_DWORD
Value	Valid values:

	- 0: Use SQL server database (default) - 1: Use Oracle database
Note	In an Oracle deployment, create the key and set the value to 1.

One Identity Manager database connection string (Q1IMDBConnectionString)

This registry setting specifies the connection string to the One Identity Manager database. This setting is used by the ManagementServer internal service that manages the core Data Governance service dependencies.

Table 100: Registry setting: Q1IMDBConnectionString

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	Q1IMDBConnectionString
Value type	REG_SZ
Value	Connection string assigned to the One Identity Manager database.

Oracle deployment: One Identity Manager database (Q1IMDBPlatformOracle)

> NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

This key defines the database system, SQL Server or Oracle Database, being used for the One Identity Manager database. This setting is used by the ManagementServer internal service that manages the core Data Governance service dependencies.

> NOTE: If you are using an Oracle database for the One Identity Manager database, create this registry key on the computer where the Data Governance service is to be installed. Once this key and the QDGDBPlatformOracle key are created and set to 1, use the Data Governance Configuration wizard to deploy the Data Governance service and create the Resource Activity database.

Table 101: Registry setting: Q1IMDBPlatformOracle

Location	Registry

Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	Q1IMDBPlatformOracle
Value type	REG_DWORD
Value	Valid values: • 0: Use SQL server database (default) • 1: Use Oracle database
Note	If using an Oracle deployment, create the key and set the value to 1.

Write default managed resource data to database (ResourceTemplateDefaultData)

This key indicates whether the default managed resource data defined in Data Governance Edition is written to the One Identity Manager database This includes the following default data about a managed resource:

- Managed group templates
- Name pattern resolvers
- Server selection scripts
- Managed resource type (that is, Simple Share)
- Type group permissions
- Managed resource functions

For more information on the managed resources, see the *One Identity Manager Data Governance Edition IT Shop Resource Access Requests User Guide*.

> NOTE: This registry value is checked on Data Governance service startup and if not present or if its value is set to 0, Data Governance Edition writes the default managed resource data into the One Identity Manager database and sets the registry value. When this value is set to 1, this indicates that the default managed resource data is already stored in One Identity Manager database and should not be overwritten on service startup.

Table 102: Registry setting: ResourceTemplateDefaultData

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server
Value name	ResourceTemplateDefaultData

Value type	REG_DWORD
Value	Valid values: • 0: Write the default managed resource data into the One Identity Manager database. • 1: Default template data is already stored in the One Identity Manager database: do not overwrite on Data Governance service startup.
Notes	If you delete the default managed resource data in your Data Governance Edition deployment and replace it with a new managed resources, you must move or set this registry key if you move the Data Governance service to another machine. When you move the Data Governance service to another machine, before starting the Data Governance service ensure that this registry key is set (value is set to 1); otherwise, the Data Governance service will reload any previously deleted template data that was inserted when the Data Governance service was initially started (on the first machine). If you modify the default managed resource data in your Data Governance Edition deployment, the data is retained if you move the Data Governance service to another machine.

Agent start/stop timeout (StartStopServiceTimeout)

Create this registry key on the Data Governance server to specify the amount of time (in seconds) to wait for an agent service to start or stop before timing out.

Table 103: Registry setting: StartStopServiceTimeout

Location	Registry
Path	HKEY_LOCAL_MACHINE\SOFTWARE\One Identity\Broadway\Server\AgentDeployment NOTE: The AgentDeployment subkey does not exist by default and will need to be created.
Value name	StartStopServiceTimeout
Value type	REG_DWORD
Value	Maximum amount of time, in seconds, before an agent start/stop times out.

	The default is 300 seconds.
Note	During an agent uninstall, the Data Governance server sends a command to stop the agent service and then waits for the service to stop. If the agent service takes a long time to shut down, the unistall fails. To solve this, the timeout value for the agent service stop can be increased using this registry key setting. In addition, if you modify the timeout value here, you should also update the RemoteExecutor.WaitResultTimeout configuration setting to the same value. For more information, see RemoteExecutor.WaitResultTimeout on page 77.

WCF timeouts (wcfTimeoutInMinutes)

Create the following registry key on the client computer where the Manager is installed to specify the maximum amount of time it should take a WCF command to complete before it times out. The Data Governance service will disconnect from the Manager client if the server does not receive any contact or messages within the allotted time.

Table 104: Registry setting: wcfTimeoutInMinutes

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client
Value name	wcfTimeoutInMinutes
Value type	DWORD
Value	Maximum amount of time, in minutes, for WCF commands to complete before timing out. Default: 5 minutes

Reporting timeout (WcfTimeoutReportingInMinutes)

Create the following registry key on the client computer where the Manager is installed to specify the maximum amount of time it should take to generate a report before it times out.

Table 105: Registry setting: WcfTimeoutReportingInMinutes

Location	Registry
Path	HKEY_CURRENT_USER\SOFTWARE\One Identity\Broadway\Client
Value name	wcfTimeoutReportingInMinutes
Value type	DWORD
Value	Maximum amount of time, in minutes, for a report to generate before timing out. Default: 15 minutes

Appendix A

PowerShell commands

This appendixprovides a list of the Windows PowerShell commands available to deploy and configure Data Governance Edition components and administer Data Governance Edition to manage the unstructured data in your organization.

- Adding the PowerShell snap-ins
- Finding component IDs
- Data Governance Edition deployment
- Service account management
- Managed domain deployment
- Agent deployment
- Managed host deployment
- Account access management
- Resource access management
- Governed data management
- Classification management

For full parameter details and examples, see the command help. For a list and full parameter details and examples of the PowerShell commands available for creating and maintaining managed resources (such as, file shares created through the IT Shop self-service request functionality), see the *One Identity Manager Data Governance Edition IT Shop Resource Access Tickets User Guide*.

Adding the PowerShell snap-ins

Data Governance Edition comes with a Windows PowerShell snap-in for you to use to manage your environment.

If you installed Windows PowerShell on your computer after you installed the Data Governance server, you must register the cmdlets before you can start using them in Windows PowerShell.

To import the Data Governance Edition 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 QAM.Client.PowerShell.dll assembly. By default, the <path> for the Data Governance server machine is "C:\Program Files\One Identity\One Identity Manager\QAM.Client.PowerShell.dll".
2. To verify that the module was added, type the following at the Windows PowerShell command prompt:

   ```
   Get-Module -All
   ```

   The registered PowerShell modules are listed.

> NOTE: Run the `Set-QServiceConnection` command before you can use any of the Data Governance Edition commands.

Adding the module automatically to new sessions

If you do not want to manually add the Data Governance Edition PowerShell module each time you start a new Windows PowerShell session, you can modify the Windows PowerShell profile file so that it is added automatically for you.

To add the Data Governance Edition PowerShell module automatically when you start a new Windows PowerShell session

- Add the following line to the Windows PowerShell profile file (profile.ps1) file:

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

  The location of the Windows PowerShell profile file is as follows: WINDOWS\system32\windowspowershell\v1.0

> NOTE: If you get the error message "...profile.ps1 cannot be loaded because the execution of scripts is disabled" the next time you start a new Windows PowerShell session, type the following at the Windows PowerShell command prompt:
>
> ```
> Set-ExecutionPolicy RemoteSigned
> ```
>
> Then, type the following at the Windows PowerShell command prompt to confirm that the execution policy has been changed:
>
> ```
> Get-ExecutionPolicy RemoteSigned
> ```

Finding component IDs

Many of the Windows PowerShell commands you can use to manipulate your deployment require that you know the component’s ID.

To determine the managed host, container parent, container, resource node, or agent ID

- Run the `Get-QManagedHosts` command.

  For more information, see Get-QManagedHosts on page 161.

To determine the service account or managed domain ID

- Run the `Get-QManagedDomains` command.

  For more information, see Get-QManagedDomains on page 141.

To determine the deployment name

- Run the `Get-QDeploymentInfo` command.

  For more information, see Get-QDeploymentInfo on page 124.

Data Governance Edition deployment

The following commands in the OneIdentity.DataGovernance snap-in can be used to deploy and configure the Data Governance Edition. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 106: Data Governance Edition deployment commands

Use this command	If you want to
Get-QDeploymentInfo	View deployment information for your Data Governance server including the deployment name. For more information, see Get-QDeploymentInfo on page 124.
Get-QEncryptionOptions	Retrieve the current encryption options used by One Identity Manager and show whether Data Governance Edition has been configured to use encryption. For more information, see Get-QEncryptionOptions on page 125.
Get-QServerAllLogs	Export all server logs to the designated folder. For more information, see Get-QServerAllLogs on page 125.
Get-ServerVersion	View the version of the currently running Data Governance server. For more information, see Get-QServerVersion on page 126.
Initialize-QDataGovernanceActivity	Initialize a database to store data generated when a managed host has resource activity tracking enabled.

Use this command	If you want to
	NOTE: This information is required for several reports, including the Resource Activity report. This is separate from the One Identity Manager database that stores configuration and security information. For more information, see Initialize-QDataGovernanceActivity on page 126.
Initialize-QDataGovernanceServer	Establish the database connection between One Identity Manager and Data Governance Edition. The Data Governance server must be initialized before you can use Data Governance Edition to manage your resources. For more information, see Initialize-QDataGovernanceServer on page 127.
Register-QServiceConnectionPoint	Register service connection points in an Active Directory domain. NOTE: This can be helpful when the service account registered for a domain does not have sufficient permissions to create a service connection point (SCP). For more information, see Register-QServiceConnectionPoint on page 129.
Remove-QServiceConnectionPoint	Remove the DataGovernance.Server Service Connection Point (SCP) from an Active Directory domain. NOTE: This cmdlet can be helpful when you want to remove all Data Governance Edition SCPs from a single Data Governance Edition deployment or all deployments. To recreate an SCP which you inadvertently removed, restart your Data Governance service. For more information, see Remove-QServiceConnectionPoint on page 130.
Set-QDeploymentInfo	Change the deployment parameters for the Data Governance server including the deployment name. NOTE: Changing this value can prevent the Data Governance service from communicating with existing agents. It is not recommended to change the deployment name of an existing server. For more information, see Set-QDeploymentInfo on page 131.
Set-QEncryptionOptions	Encrypt the Data Governance service account. NOTE: Only use this command if you have enabled encryption for the One Identity Manager database.

Use this command	If you want to
	For more information, see Set-QEncryptionOptions on page 132.
Set-QServiceConnection	Set the server name and port information used by the Data Governance Edition commands to connect to the Data Governance server. NOTE: You must run this command before you can use any of the Data Governance Edition commands. For more information, see Set-QServiceConnection on page 132.

Get-QDeploymentInfo

Retrieves deployment parameters, including the deployment name, for a Data Governance server.

Syntax:

```
Get-QDeploymentInfo [<CommonParameters>]
```

Examples:

Table 107: Examples

Example	Description
Get-QDeploymentInfo	Returns the current deployment parameters for the Data Governance server.

Details retrieved:

Table 108: Details retrieved

Detail	Description
DeploymentId	Name assigned to the deployment when the Data Governance server was installed. The default deployment name is "DEFAULT".
RestServicePort	Port used by the Data Governance server for HTTP protocol and REST services. Used for communication with PowerShell and One Identity Manager clients and web server.
DatabaseMigrationVersion	The module and migration version assigned to the QAM module.

Get-QEncryptionOptions

Retrieves the current encryption options that One Identity Manager uses and indicates whether Data Governance Edition has been configured to use encryption.

Syntax:

```
Get-QEncryptionOptions [<CommonParameters>]
```

Example

Table 109: Examples

Example	Description
Get-QEncryptionOptions	Retrieves the current encryption information.

Details retrieved

Table 110: Details retrieved

Detail	Description
IsDGEConfigured	Indicates whether Data Governance Edition is configured to use encryption.
Scheme	The algorithm currently being used for One Identity Manager encryption.

Get-QServerAllLogs

Exports all server logs (the DataGovernanceEdition.Service.exe.dlog file and associated agent deployment logs) to the designated folder.

Syntax:

```
Get-QServerAllLogs -OutputFolder <String> [<CommonParameters>]
```

Table 111: Parameters

Parameter	Description
OutputFoler	Specify the folder where the logs are to be saved. NOTE: The output folder must already exist on the Data Governance server.

Examples:

Table 112: Examples

Example	Description
Get-QServerAllLogs -OutputFolder D:\ServerLogs	Exports all server logs to the designated location (D:\ServerLogs in this example).

Get-QServerVersion

Returns the version of the currently running Data Governance server.

Syntax:

```
Get-QServerVersion [<CommonParameters>]
```

Examples:

Table 113: Examples

Example	Description
Get-QServerVersion	Returns current version of the Data Governance server.

Details retrieved:

Table 114: Details retrieved

Detail	Description
Major	The major version number.
Minor	The minor version number.
Build	The build number.
Revision	The revision number.

Initialize-QDataGovernanceActivity

Initializes a database to store data generated when a managed host has resource activity tracking enabled.

NOTE: Resource tracking activity is required for several reports, including the Resource Activity report. This database is only for audit information; it is separate from the One Identity Manager database which store configuration and security information.

Syntax:

```
Initialize-QDataGovernanceActivity [-ConnectionString] <String> [[-Validate]
[<SwithParameter>]] [[-ActivityDatabaseIsOracle [<SwitchParameter>]]
[<CommonParameters>]
```

Table 115: Parameters

Parameter	Description
ConnectionString	Specify the database connection string used by Data Governance Edition to access the Resource Activity database.
Validate	Specify this parameter if you want to the cmdlet to validate the connection string and fail if is in not valid.
ActivityDatabaseIsOracle	If you are using an Oracle database management system for the Resource Activity database, specify this parameter to indicate that it is an Oracle database. NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.

Examples:

Table 116: Examples

Example	Description
Initialize -QDataGovernanceActivity -ConnectionString "Data Source=QAMDB;Initial Catalog=QAMAUDITDB;User ID=sa;Password=template$PWD" -Validate	Sets the connection string for the Resource Activity database; validating the connection string before proceeding.

Initialize-QDataGovernanceServer

Establishes the database connection between One Identity Manager and Data Governance Edition. The Data Governance server must be initialized before you can use Data Governance Edition to manage your resources.

NOTE: This PowerShell cmdlet is used in conjunction with the Data Governance Server installation msi when manually installing Data Governance Edition.

Syntax:

```
Initialize-QDataGovernanceServer [-DatabaseConnectionString] <String> [[-
IdentityManagerIsOracle [<SwitchParameter>]] [-DefaultEmployeeSid [<String>]]
[<CommonParameters>]
```

Table 117: Parameters

Parameter	Description
DatabaseConnectionString	Specify the database connection string used by Data Governance Edition to access the One Identity Manager database. An example of a connection string for Windows authentication may look like this: "Server=myServerAddress;Database=myDatabase;User Id=myUser;Password=myPassword;Trusted_Connection=True" An example of a connection string for SQL authentication may look like this: "Data Source=myServerAddress;Initial Catalog=myDatabase;User Id=myUser;Password=myPassword"
IdentityManagerIsOracle	If you are using an Oracle database management system for the One Identity Manager database, specify this parameter to indicate that an Oracle database is being used. NOTE: Oracle Database support was deprecated beginning with One Identity Manager 8.1. Do not use.
DefaultEmployeeSid	(Optional) Specify this parameter to take advantage of the automatic forest topology harvest. That is, adding this parameter adds the user associated with the specified SID to the One Identity Manager Identities with appropriate Data Governance application roles. NOTE: This provides the same functionality as selecting the **Add the current user to the One Identity Manager Identities with Data Governance application roles** option when using the Data Governance Configuration wizard.

Examples:

Table 118: Examples

Example	Description
Initialize-QDataGovernanceServer -	Initializes Data Governance

Example	Description
DatabaseConnectionString 'Data Source=IMSQL;Initial Catalog=OneIM;UserID=sa;Password=template$PWD'	Edition with the One Identity Manager database with the supplied connection string
Initialize-QDataGovernanceServer -DatabaseConnectionString 'Data Source=IMSQL;Initial Catalog=OneIM;UserID=sa;Password=myPwd' -DefaultEmployeeSid S-1-5-21-2969523365-1970145350-1015297841-500'	Establishes connection between the One Identity Manager database and Data Governance Edition; and adds the specified identity to the One Identity Manager Identities with the Data Governance application roles.

Register-QServiceConnectionPoint

Registers service connection points (SCPs) in an Active Directory domain.

NOTE: This can be helpful when the service account registered to a domain does not have sufficient permissions to create an SCP.

Syntax:

```
Register-QServiceConnectionPoint [-DomainDnsName] <String> [-DeploymentId]
<String> [-ServerDnsName] <String> [[-ServerNetTcpPortNumber] [>Int32>]]
[<CommonParameters>]
```

Table 119: Parameters

Parameter	Description
DomainDnsName	Specify the full DNS name of the Active Directory domain where the SCP will be registered.
DeploymentId	Specify the deployment name of the Data Governance instance.
ServerDnsName	Specify the full DNS name of the computer hosting the Data Governance server.
ServerNetTcpPortNumber	(Optional) Specify the Net.tcp port number of the Data Governance server. If this parameter is not specified, the default port (8722) is used.

Examples:

Table 120: Examples

Example	Description
Register-QServiceConnectionPoint -DomainDnsName vmset6.dge.dev.ca -DeploymentId DGEMAIN - ServerDnsName 2k8.vmset6.dge.dev.ca	Registers the SCP for an Active Directory domain.

Remove-QServiceConnectionPoint

Removes DataGovernance.Server service connection points (SCPs) found by the global catalog (GC) search in the Active Directory forest.

> NOTE: This cmdlet can be helpful when you want to remove all Data Governance Edition SCPs from a single Data Governance Edition deployment or all deployments.
>
> To re-create an SCP which you inadvertently removed, restart your Data Governance server.

Syntax:

```
Remove-QServiceConnectionPoint [-DeploymentId] [<String>]]
[<CommonParameters>]
```

Table 121: Parameters

Parameter	Description
DeploymentId	(Optional) Specify the deployment name assigned to the Data Governance instance whose SCP is to be removed. Typically, this value is DEFAULT. If you do not specify this parameter, all service connection points in all Data Governance Edition deployments will be removed. Run the Get-QDeploymentInfo cmdlet to retrieve the deployment name (DeploymentId) assigned to a Data Governance Edition deployment.

Examples:

Table 122: Examples

Example	Description
Remove-QServiceConnectionPoint	Removes all service connection points in all deployments.

Example	Description
Remove-QServiceConnectionPoint -DeploymentId MyTestDeployment	Removes all service connection points in the Data Governance instance assigned to deployment "MyTestDeployment".

Set-QDeploymentInfo

Updates the deployment parameters for the Data Governance server.

> NOTE: Changing the deployment identifier parameter can prevent the Data Governance service from communicating with existing agents. It is NOT recommended to change the deployment name of an existing Data Governance server.

Syntax:

```
Set-QDeploymentInfo [-DeploymentId [<String>]] [<CommonParameters>]
```

Table 123: Parameters

Parameter	Description
DeploymentId	(Optional) Specify this parameter to change the name of the deployment to which this Data Governance server belongs. This deployment name must be unique within your Active Directory forest. It has a maximum length of 30 characters; and can only contain alphanumeric characters and underscores (no spaces allowed). IMPORTANT: This is NOT the recommended approach. It is best to uninstall your entire Data Governance Edition deployment and reinstall using the Data Governance Configuration wizard, which comes with the One Identity Manager installation. IMPORTANT: Any existing agents deployed by this Data Governance server will not be able to connect if you change the deployment name here. You must first uninstall all the agents and then change this value, restart the DataGovernance.Server service, and deploy new agents.

Examples:

Table 124: Examples

Example	Description
Set-QDeploymentInfo -DeploymentId MainDeployment	Changes the deployment name assigned to the Data Governance service to "MainDeployment".

Set-QEncryptionOptions

Encrypts the Data Governance service account.

> NOTE: Only use this cmdlet if you have enabled encryption for the One Identity Manager database.

Syntax:

Set-QEncryptionOptions [-File] <String> [[-FIPSCompliantRSA] [<SwitchParameter>]] [[-RSA] [<SwitchParameter>]] [<CommonParameters>]

Table 125: Parameters

Parameter	Description
File	Specify the path to the file that contains the encryption key information.
FIPSCompliantRSA	(Optional) Specify this parameter if FIPS compliant algorithm will be used.
RSA	(Optional) Specify this parameter if RSA compliant algorithm will be used.

Examples:

Table 126: Examples

Example	Description
Set-QEncryptionOptions -File \\2k8R2DJSQL\C$\key -RSA	Encrypts the Data Governance service account using RSA compliant algorithm.

Set-QServiceConnection

Sets the deployment name, server name and port information used by the Data Governance Edition commands to connect to the Data Governance server.

> NOTE: This cmdlet must run before you can use any of the Data Governance Edition commands.

Syntax:

Set-QServiceConnection [-DeploymentId] [<String>]] [-ServerName [<String>]] [-Port [<String>]] [-Validate [<SwitchParameter>]] [<CommonParameters>]

Table 127: Parameters

Parameter	Description
DeploymentId	(Optional) Specify the deployment name of the Data Governance Edition deployment you wish to connect. If you are unsure of the deployment name, specify the server name (-ServerName parameter).
ServerName	(Optional) Specify the name of the server to be used by the Data Governance Edition commands. This can be specified in DNS, pre-Windows 2000 or IP address format. If you are unsure of the server name, specify the deployment name (-DeploymentId parameter).
Port	(Optional) Specify the listening port in the Data Governance Edition service configuration. If this parameter is not specified, the default port (8722) is used. If you are unsure of the port number, specify the deployment name (-DeploymentId parameter).
Validate	(Optional) Specify this parameter to change the flag that indicates whether to validate the connection.

Examples:

Table 128: Examples

Example	Description
Set-QServiceConnection -ServerName qamautomem1 -Port 8722	Sets the service connection for a server on the computer named qamautomem1 on port 8722.
Set-QServiceConnection -DeploymentId MainDeployment	Sets the service connection for a server in the MainDeployment deployment.

Service account management

Data Governance Edition consolidates security information across many domains and forests by accessing these network entities using stored credentials (service accounts). These service accounts are Active Directory users granted the appropriate permissions in their respective domains and registered with Data Governance Edition.

The following commands are available to you to manage service accounts. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 129: Service account management commands

Use this command	If you want to
Add-QServiceAccount	Register an account as a service account for Data Governance Edition. When you add this service account, it is automatically granted the required Log On as a Service local user right on the Data Governance server. For more information, see Add-QServiceAccount on page 134.
Get-QLogonServiceAccount	Determine if the account can be used as a service account. For more information, see Get-QLogonServiceAccount on page 136.
Get-QServiceAccounts	View a list of service accounts that have been created for the Data Governance server. NOTE: You can optionally specify a service account id if you are only interested in a particular service account. For more information, see Get-QServiceAccounts on page 136.
Remove-QServiceAccount	Remove a service account from the deployment. NOTE: Remove any associated managed domains BEFORE removing a service account. For more information, see Remove-QServiceAccount on page 138.
Set-QServiceAccountUpdated	Have the Data Governance server update a service account. For more information, see Set-QServiceAccountUpdated on page 139.

Add-QServiceAccount

Registers an account as a service account for Data Governance Edition. When you add this service account, it is automatically granted the required Log On as a Service local user rights on the Data Governance server.

Data Governance Edition consolidates security information across many domains and forests by accessing these network entities using stored credentials (Service Accounts). These Service Accounts are Active Directory users granted the appropriate permissions in their respective domains and registered with Data Governance Edition.

The Service Account performs actions that a local service cannot. For example, a remote agent needs a Service Account to access the files on the managed host it is scanning.

> NOTE: Service Accounts must have administrative privileges in the domains they are registered with. This allows the Data Governance server to elevate its identity to these

accounts and perform actions such as agent deployments and Active Directory queries.

Syntax:

```
Add-QServiceAccount [-AccountDomain] <String> [-AccountName] <String> [-
Password] <String> [[-IsDefaultObjectResolution] [<Boolean>]]
[<CommonParameters>]
```

Table 130: Parameters

Parameter	Description
AccountDomain	Specify the pre-Windows 2000 name of the account domain.
AccountName	Specify the logon name (pre-Windows 2000 name) of the account.
Password	Specify the password associated with the account.
IsDefaultObjectResolution	(Optional) Specify this parameter to indicate whether the account being registered is to be used as the Data Governance default account. This account will be used to connect to Active Directories which do not have explicit service accounts configured. Valid values are: • 0 or $false: The account is not used as the Data Governance default account (default). • 1 or $true: The account is used as the Data Governance default account.

Examples:

Table 131: Examples

Example	Description
Add-QServiceAccount -AccountDomain "qamauto" -AccountName "administrator" -Password 'Pa$$word'	Adds a service account for the domain "qamauto", with the user name of "administrator" and a password of 'Pa$$word'. NOTE: Single quotes are used around the password text because it contains $ characters.

Get-QLogonServiceAccount

Determines if the specified account meets the requirements to be used as a service account in Data Governance Edition.

> NOTE:Data Governance Edition consolidates security information across many domains and forests by accessing these network entities using stored credentials (service accounts). These service accounts are Active Directory users granted the appropriate permissions in their respective domains and registered with Data Governance Edition.

Syntax:

```
Get-QLogonServiceAccount [-UserName] <String> [-Password] <String> [-DomainId] <String> [<CommonParameters>]
```

Table 132: Parameters

Parameter	Description
UserName	Specify the name of the Active Directory account to be checked.
Password	Specify the password associated with the account.
DomainName	Specify the name of the domain to be checked to determine if the specified account meets the requirements of a service account.

Examples:

Table 133: Examples

Example	Description
Get-QLogonServiceAccount -UserName Administrator -Password myppassword -DomainName mydomain.dge.dev.phx.com	Checks the specified account to determine if it meets the requirements to be used as a service account in Data Governance Edition.

Get-QServiceAccounts

Retrieves a list of service accounts registered with the Data Governance server.

Syntax:

```
Get-QServiceAccounts [-ServiceAccountId] [<String>]] [<CommonParameters>]
```

Table 134: Parameters

Parameter	Description
ServiceAccountId	(Optional) Specify the ID (GUID format) of the service account to be retrieved. Run the Get-QManagedDomains cmdlet to retrieve a list of managed domains, including the managed domain and service account IDs.

Examples:

Table 135: Examples

Example	Description
Get-QServiceAccounts	Retrieves a list of all registered service accounts.
Get-QServiceAccounts -ServiceAccountId 3253af66-c104-4472-b770-c8097b2df6d8	Retrieves information about the specified service account.

Details retrieved:

Table 136: Details retrieved

Detail	Description (Associated key or property in QAMServiceAccount table)
ServiceAccountId	The value (GUID) assigned to the service account (UID_QAMServiceAccount).
AccountSid	The security identifier (SID) assigned to the Active Directory account.
UserDomainName	The name of the domain to which the user belongs.
UserName	Logon name (pre-Windows 2000) of the Active Directory account (UID_ADSAccount).
UserPrincipalName	User principal name (email address) of the service account.
Description	The descriptive text entered when the service account was registered with Data Governance Edition.
IsDefaultObjectResolution	Indicates whether the account is being used as the Data Governance default account and will be used to connect to Active Directories which do not have explicit service accounts configured.
StatusDetailMessage	If applicable, a message about the current state of the data from the agent.

Detail	Description (Associated key or property in QAMServiceAccount table)
Status	The status of the agent.
CanManageDomains	Indicates whether the service account is capable of being impersonated on the Management Server it is being called upon. NOTE: This is set within the ServiceAccounts InternalService on the Data Governance server. It will be true if impersonation is successful; and false, if impersonation fails.
ServiceAccountName	The name of the service account.

Remove-QServiceAccount

Removes a server account from the Data Governance Edition deployment.

> NOTE: Remove any associated managed domains BEFORE removing a service account. Run the Remove-QManagedDomain cmdlet to remove a managed domain from your Data Governance Edition deployment.

Syntax:

```
Remove-QServiceAccount [-ServiceAccountId] <String> [<CommonParameters>]
```

Table 137: Parameters

Parameter	Description
ServiceAccountId	Specify the ID (GUID format) of the server account to be removed from the list of registered service accounts. Run the Get-QServiceAccounts cmdlet without any parameters to retrieve a list of registered service accounts, including the assigned service account ID.

Examples:

Table 138: Examples

Example	Description
Remove-QServiceAccount -ServiceAccountId b0a0e218-55c1-41d7-9585-bf7578ad1130	Removes the specified service account from the list of service accounts registered for use by Data Governance Edition.

Set-QServiceAccountUpdated

Notifies the Data Governance server that the service account was updated and the server should process it.

Syntax:

```
Set-QServiceAccountUpdated [-ServiceAccountId] <String>
[<CommonParameters>]
```

Table 139: Parameters

Parameter	Description
ServiceAccountId	Specify the id of the service account to be updated. Run the Get-QManagedDomains and Get-QServiceAccounts cmdlets to retrieve a list of available service accounts and their IDs.

Examples:

Table 140: Examples

Example	Description
Set-QServiceAccountUpdated -ServiceAccountId 18CC36D3-81AE-4856-925B-9B1B1E587381	Updates the specified service account.

Managed domain deployment

Before you can gather information on the data in your enterprise, you must specify the domain that contains the computers and data that you want to manage. Then assign the service account to access the resources within them.

The following commands are available to you to deploy managed domains. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 141: Managed domain deployment commands

Use this command	If you want to
Add-QManagedDomain	Add a new domain to the Data Governance Edition deployment. For more information, see Add-QManagedDomain on page 140.
Get-QManagedDomains	View the list of managed domains in a deployment.

Use this command	If you want to
	NOTE: You can optionally specify a managed domain ID if you are only interested in a particular domain. For more information, see Get-QManagedDomains on page 141.
Remove-QManagedDomain	Remove a managed domain from your deployment. For more information, see Remove-QManagedDomain on page 142.

Add-QManagedDomain

Adds a new domain to the Data Governance Edition deployment.

The Data Governance server constructs an in-memory map of the Active Directory forest and domain structure where it is deployed. Administrators responsible for the Data Governance Edition deployment must register Service Accounts with the system and link them with domains. The link between a Service Account and an Active Directory domain makes it a "managed domain".

> NOTE: Only domains that have been previously synchronized into the One Identity Manager database are available to be managed by Data Governance Edition.

Syntax:

```
Add-QManagedDomain [-ServiceAccountID] <String> [-DomainName] <String>
[<CommonParameters>]
```

Table 142: Parameters

Parameter	Description
ServiceAccountID	Specify the ID (GUID format) of the service account that will manage the domain. Run the Get-QServiceAccounts cmdlet to retrieve a list of all service accounts registered with your Data Governance Edition deployment.
DomainName	Specify the DNS name of the domain to be added as a managed domain.

Examples:

Table 143: Examples

Example	Description
Add-QManagedDomain -ServiceAccountID 7dd2eb51-e1cb-47f2-8c76-093fd4e0459e -DomainName mydomain.local	Adds a new managed domain.

Get-QManagedDomains

Retrieves information, including the service account and managed domain IDs, for a managed domain from the Data Governance Edition deployment.

Syntax:

```
Get-QManagedDomains [-ManagedDomainId [<String>]] [<CommonParameters>]
```

Table 144: Parameters

Parameter	Description
ManagedDomainId	(Optional) Specify the ID (GUID format) of the managed domain to be retrieved.

Examples:

Table 145: Examples

Example	Description
Get-QManagedDomains	Returns all managed domains in the database.
Get-QManagedDomains -ManagedDomainId 50905871-5379-455d-8b65-c4bd02360bdb	Returns information on the specified managed domain.

Details retrieved:

Table 146: Details retrieved

Detail	Description (Associated key or property in ADSDomain table)
ManagedDomainID	The value (GUID) assigned to the managed domain. (UID_ADSDomain)

Detail	Description (Associated key or property in ADSDomain table)
DomainDnsName	The full DNS name of the managed domain. (ADSDomainName)
ForestDnsName	The full DNS name of the forest where the domain resides. (UID_ADSForest)
Status	The status of the managed host, based on all the agents monitoring the host.
NetbiosName	The Netbios name of the managed domain.
DomainSid	The security identifier (SID) assigned to the managed domain.
ServiceAccountId	The value (GUID) of the service account assigned to the managed domain. (UID_QAMServiceAccount)
AccessGroupSid	Deprecated.
ServiceAccountInfo	The name of the service account assigned to the managed domain.
DomainControllerName	The name of the domain controller hosting the managed domain.
ExtendedRightsCreated	Indicates whether extended rights were created by Data Governance Edition in the Active Directory environment.

Remove-QManagedDomain

Removes a managed domain from the Data Governance Edition deployment.

> NOTE: Remove all managed hosts associated with a managed domain BEFORE removing a managed domain. Run the Remove-QManagedHost cmdlet to remove a managed host.

Syntax:

```
Remove-QManagedDomain [-ManagedDomainId] <String> [<CommonParameters>]
```

Table 147: Parameters

Parameter	Description
ManagedDomainId	Specify the ID (GUID format) of the managed domain to be removed. Run the Get-QManagedDomains cmdlet without any parameters to retrieve a list of managed domains, including the managed domain ID.

Examples:

Table 148: Examples

Example	Description
Remove-QManagedDomain -ManagedDomainId 830b1e48-c682-4d3e-965c-d96ee6db6262	Removes the specified managed domain from Data Governance Edition.

Agent deployment

The following commands are available to you to manage your agent deployment. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 149: Agent deployment commands

Use this command	If you want to
Get-QAgentEvents	View saved events for the specified agent from the One Identity Manager database. You can use this command to output the stored agent messages to the console or a text file to quickly identify issues. For more information, see Get-QAgentEvents on page 144.
Get-QAgentMetrics	View an agent's activity and performance. For more information, see Get-QAgentMetrics on page 145.
Set-QAgentConfiguration	Set the managed paths to be scanned. NOTE: When you set the managed paths using the cmdlet, existing managed paths are overwritten. NOTE: This cmdlet does not support setting managed paths for Cloud managed hosts. For more information, see Set-QAgentConfiguration on page 146.
Set-QAgentStateUpdated	Notify the Data Governance server that an agent has been updated and the server should process it. For more information, see Set-QAgentStateUpdated on page 147.
Upgrade-QAgents	Upgrade the agents in your deployment. NOTE: You can identify the agents to upgrade through their agent ID or on a managed host basis. For more information, see Upgrade-QAgents on page 148.

Get-QAgentEvents

Retrieves saved events for the specified agent from the One Identity Manager database.

Agent events include information such as errors and warnings from the agent that are stored in the One Identity Manager database. Use this command to retrieve stored agent messages to the console or a text file to quickly identify issues.

Syntax:

```
Get-QAgentEvents [-AgentId] <String> [<CommonParameters>]
```

Table 150: Parameters

Parameter	Description
AgentId	Specify the ID (GUID format) of the agent whose stored messages are to be retrieved. Run the Get-QManagedHosts cmdlet and locate the agents array in the managed host to retrieve its ID.

Examples:

Table 151: Examples

Example	Description
C:\PS>@{#cdata-section=Add-PSSnapin Quest.DataGovernance $hosts = Get-QManagedHosts $agents = $hosts[0].Agents $agentId = $agents[0].Id Get-QAgentEvents -AgentId $agentId; xmlns:dev=http://schemas.microsoft.com/maml/dev/2004/10} First run the Get-QManagedHosts command and access the agents collection for the selected managed host. Locate the agent ID, call Get-QAgentEvents, and pass the agent ID to the method.	Returns all stored messages for the specified Data Governance agent.

Details retrieved:

Table 152: Details retrieved

Detail	Description (Associated key or property in QAMAgentEvent table)
Date/Time	The date and time the critical agent error occurred.
Description	The event message logged for the critical agent error.

Get-QAgentMetrics

Retrieves agent activity and performance metrics.

Syntax:

```
Get-QAgentMetrics [-AgentId] <String> [<CommonParameters>]
```

Table 153: Parameters

Parameter	Description
AgentId	Specify the ID (GUID format) of the agent whose metrics are to be retrieved. Run the Get-QManagedHosts cmdlet and locate the agents array in the managed host to retrieve its ID.

Examples:

Table 154: Examples

Example	Description
(Get-QAgentMetrics -AgentId 19048a06-845a-4628-94fc-dabf60345ea9).Metrics	Returns agent activity and performance information for the specified Data Governance agent. Expands the Metrics array to display the individual metrics.

Details retrieved:

Table 155: Details retrieved

Detail	Description
Agent	Value (GUID) assigned to the agent.
MetricsSetName	The name associated with a set of metrics: • AgentCore • Resource Activity • Scan • SharePoint Security Indexer • SharePoint Security Indexer - Store Security Info
Start	For metrics that span a length of time, the date and time when the metrics collection started.

Detail	Description
End	For metrics that span a length of time, the date and time when the metrics collection ended. If no end time is specified by the agent, the date/time maximum value (for example, 23:59:59:9999999 UTC, December 31, 9999)
Metrics	Metrics is an array that can be expanded to show the metrics returned from the agent. The following details are displayed for each metric: • Value • ValueAsObject • ValueAsString • Name

Set-QAgentConfiguration

Sets or modifies the managed paths to be scanned by the specified agent.

NOTE: When you set the managed paths using this cmdlet, existing managed paths will be overwritten.

NOTE: This cmdlet is does not support setting managed paths for Cloud managed hosts.

Syntax:

```
Set-QAgentConfiguration [-DataRoots [<String[]>]] [-AgentId] <String> [-
ManagedHostId] <String> [-AppendRoots [<SwitchParameter>]]
[<CommonParameters>]
```

Table 156: Parameters

Parameter	Description
DataRoots	(Optional) Specify this parameter to specify or change the managed paths to be scanned by the agent. Enter an array of strings that contain the paths to the roots to be scanned by the agent. For SharePoint managed host, enter the DisplayPath|UnfriendlyPath (see example).
AgentId	Specify the ID (GUID format) of the agent you want to set (or change) roots for. Run the Get-QManagedHosts cmdlet and locate the agents array in the managed host to retrieve its ID.
ManagedHostId	Specify the ID (GUID format) of the managed host you want to set (or

Parameter	Description
	change) managed paths for. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
AppendRoots	(Optional) Specify this parameter to append any specified managed paths (DataRoots) to the given agent configuration. If not specified, the data roots provided overwrite previously defined managed paths.

Examples:

Table 157: Examples

Example	Description
Set-QAgentConfiguration -AgentId 49536bfa-d149-4410-a484-aca06dbef09e -ManagedHostId fef62b17-146b-4eb3-9567-7707b6a88785 -DataRoots \\2K8R2DJSQL\C$\Test Data	Sets the managed paths to be scanned for the specified agent.
Set-QAgentConfiguration -AgentId 67e1c215-6603-42f5-b5a1-42a05837ae12 -ManagedHostId 8212e02a-7b3f-4010-bb33-68160abc41fc -DataRoots "SharePoint_ConfigVmset6/SharePoint - 80/My Wiki/My Wiki/Documents|sp://titan/0ee296d6-dea5-4f4d-950f-27c06458cad1/57947f70-c2b0-4d76-a8b3-ac54fa5bb4ab/15c4fc23-b986-4937-890c-d38715d3114/My%20Wiki/Documents"	Specifies the managed paths to be scanned for a SharePoint managed host. TIP: The DataRoots is (DisplayPath|Un-friendlyPath).

Set-QAgentStateUpdated

Notifies the Data Governance server that an agent has been updated and the server should process it

NOTE: Only use this cmdlet if you have enabled encryption for the One Identity Manager database.

Syntax:

```
Set-QAgentStateUpdated [-AgentId] <String> [<CommonParameters>]
```

Table 158: Parameters

Parameter	Description
AgentId	Specify the ID (GUID format) of the agent that was updated. Run the Get-QManagedHosts cmdlet and locate the agents array in the managed host to retrieve its ID.

Examples:

Table 159: Examples

Example	Description
Set-QAgentStateUpdated -AgentId 37b27a56-3463-45a8-83ec-ff88d48c49a7	Tells the Data Governance server to update the agent state for the specified agent.

Upgrade-QAgents

Upgrades the agents in your Data Governance Edition deployment.

Syntax:

```
Upgrade-QAgents [-ManagedHostIds] <String[]> [-AgentIds] <String[]> [-
UpgradeAllAgents] <SwitchParameter. [[-BatchSize] [<Int32>]] [[-OnErrorContinue
[ [<SwitchParameter>]] [<CommonParameters>]
```

Table 160: Parameters

Parameter	Description
ManagedHostIds	Specify the ID (GUID format) of the managed hosts to have their agents upgraded. This parameter supports a single value, a list of comma-separated values, or an array. NOTE: Do not specify this parameter with the -AgentIds or -UpgradeAllAgents parameter. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
AgentIds	Specify the ID (GUID format) of the agents to be upgraded. This parameter supports a single value, a list of comma-separated values, or an array. Run the Get-QManagedHosts cmdlet and locate the agents array in the managed host to retrieve its ID.

Parameter	Description
	NOTE: Do not specify this parameter with the -ManagedHostIds or -UpgradeAllAgents parameter. NOTE: Upgrading an agent on an agent host will cause all agents on the same server to be upgraded.
UpdateAllAgents	Specify this parameter to upgrade all of the agents that are in an upgradable state. NOTE: Do not specify this parameter with the -ManagedHostIds or -AgentIds parameter.
BatchSize	(Optional) Specify the batch size of the agents/managed hosts that are able to be updated simultaneously. Each host/agent in the batch must complete their upgrade (or fail) before the next batch of hosts/agents are processed. If this parameter is not specified, the default batch size of five is used. NOTE: If a host/agent in the current batch fails, the upgrade process will be aborted for all queued machines unless the -OnErrorContinue flag is set.
OnErrorContinue	(Optional) Specify this parameter to allow subsequent batches of hosts/agents to be processed for upgrade even if the upgrade of a host/agent in the previous batch failed.

Examples:

Table 161: Examples

Example	Description
Upgrade-QAgents -UpgradeAllAgents -OnErrorContinue	Upgrades all agents, in batches of five (default); processing upgrades even if an agent in the previous batch has failed.
Upgrade-QAgents -ManagedHostIds ("0f04f33e-6717-4cfc-8528-9e396137f-d6e","0f04f33e-6717-4cfc-8528-9e396137f-d6e") -BatchSize 7 -OnErrorContinue	Updates all agents associated with the two specified managed hosts in batches of seven; processing upgrades even if an agent in the previous batch has failed.

Managed host deployment

A managed host is any network object that can host resources and can be assigned an agent to monitor security and resource activity. Currently supported hosts include Windows computers, Windows clusters, NetApp storage devices, EMC storage devices, DFS, and SharePoint farms.

You can also add generic managed hosts (Server Message Block (SMB) shares running on any Active Directory joined computer) to remotely scan their resources.

The following commands are available to you to deploy managed hosts. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 162: Managed host deployment commands

Use this command	If you want to
Add-QDfsManagedHost	Register a domain-based distributed file system root. This enables you to view and manage the access on resources that are physically distributed throughout your network. For more information, see Add-QDfsManagedHost on page 151.
Add-QManagedHostByAccountName	Add a managed host to your deployment and configure its settings. For more information, see Add-QManagedHostByAccountName on page 152. NOTE: This cmdlet does not support adding Cloud managed hosts.
Clear-QResourceActivity	Clear the resource activity for a given managed host. This enables you to remove activity data from the database on demand when it is no longer required. For scheduled activity cleanup, use the activity compression/deletion settings in the Data Governance server configuration file instead. NOTE: Once you clear the activity, it cannot be recovered. For more information, see Clear-QResourceActivity on page 159.
Get-QHostsforTrustee	View a selected user or group’s access on all managed hosts in your environment. For more information, see Get-QHostsForTrustee on page 160.
Get-QManagedHosts	View a list of all the managed hosts in your deployment. NOTE: If you are interested in only one managed host, you can specify the host's name or the ID (GUID format) of the managed host. You can also specify all the managed hosts in a particular

Use this command	If you want to
	container. For more information, see Get-QManagedHosts on page 161.
Remove-QManagedHost	Remove a managed host from your deployment. For more information, see Remove-QManagedHost on page 164.
Set-QManagedHostProperties	Change the properties of a managed host. NOTE: You must know the managed host ID For more information, see Set-QManagedHostProperties on page 165.
Set-QManagedHostUpdated	Inform the Data Governance server that the managed host state should be updated. For more information, see Set-QManagedHostUpdated on page 169.
Trigger-QDfsSync	By default the Data Governance server synchronizes the DFS structure into the One Identity Manager database every 24 hours. Use this cmdlet to force a DFS synchronization of a DFS managed host, making the DFS path immediately available within the Resource browser. NOTE: You must specify the ID (GUID format) of the managed host to be synchronized. To synchronize all of the DFS managed hosts in your deployment, set the ManagedHostID to All. For more information, see Trigger-QDfsSync on page 170.

Add-QDfsManagedHost

Registers a domain-based distributed file system (DFS) root with Data Governance Edition. This enables you to view and manage the access on resources that are physically distributed throughout the network. Once added, the Data Governance server periodically synchronizes the DFS structure into the One Identity Manager database making the DFS path available within the Resource browser.

NOTE: The domain specified must be managed.

Syntax:

```
Add-QDfsManagedHost [-ManagedDomain] <String> [-DfsRoot] <String>
[<CommonParameters>]
```

Table 163: Parameters

Parameter	Description
ManagedDomain	Specify the NetBIOS or DNS name of a managed domain. Run the Get-QManagedDomains cmdlet to retrieve a list of all managed domains in your Data Governance Edition deployment.
DfsRoot	Specify the name of the distributed file system root in the managed domain.

Examples:

Table 164: Examples

Example	Description
Add-QDfsManagedHost --ManagedDomain 'anchor.acme.com' -DfsRoot 'software'	Registers the domain-based distributed file system root "\\anchor.acme.-com\software", where "anchor.acme.com" is the domain and "software" is the DFS root.

Add-QManagedHostByAccountName

Registers a computer as a managed host with your Data Governance Edition deployment and configures its settings.

A managed host is any network objects that can host resources and can be assigned an agent to monitor security and collect resource activity. Currently supported hosts include:

- Local Windows computer
- Windows Cluster/Remote Windows computer
- Generic resource (that is, a Server Message Block (SMB) share running on any Active Directory joined computer)
- Distributed File System (DFS) root
- SharePoint farm
- EMC storage device with CIFS file system protocol enabled
- NetApp 7-Mode filer with CIFS file system protocol enabled
- NetApp Cluster-Mode filer with CIFS file system protocol enabled

- EMC Isilon storage device with NFS system protocol enabled
- NetApp 7-Mode filer with NFS file system protocol enabled
- NetApp Cluster-Mode filer with NFS file system protocol enabled

> NOTE: This PowerShell cmdlet does not support adding Cloud managed hosts.

Once you have added a managed host, you can begin to manage the data contained within it.

Syntax:

```
Add-QManagedHostByAccountName [-HostAccountName] <String[]> [[-Keyword]
[<String>]] [[-ResourceActivityEnabled] [<SwitchParameter]] [[-Granularity
[<Int32>]] [[-ExcludedTrusteesImportFile [<String>]] [[-
ExcludedFileTypesImportFile] [<String>]] [[-ExcludedFoldersImportFile]
[<String>]] [[-AgentHostName] [<String>]] [[-SelectedDataRoots] [<String>]] [[-
ScheduleType]
[<QAM.Common.Interfaces.ScheduleConfiguration+ScanScheduleType>]] [[-
RunOnDays] [<String>]] [[-ScheduledTime] [<String>]] [[-ScanInterval]
[<Int32>]] [[-ServiceAccountId] [<String>]]
[[EnableRemoteFileSystemChangeWatching] [<SwitchParameter>]] [[-
PerformImmediateScanOnWatchError] [<SwitchParameter.]] [[-
OverrideScanScheduleOnStartup] [<Boolean>]] [[-HostType]
[<QAM.Common.Interfaces.ManagedHostInfo+HostTypes>]] [-DataRootType
[<String>]] [[-IsManagedResourceHost] [<Boolean>]] [[-IgnoreFiles]
[<SwitchParameter>]] [<CommonParameters>]
```

Table 165: Parameters

Parameter	Description
HostAccountName	Specify the managed host account name.
Keyword	(Optional) Specify a keyword that can be used to group managed hosts in the **Managed host** view of the Manager.
ResourceActivityEnabled	(Optional) Specify this parameter to enable resource activity collection. Resource activity collection is disabled by default. You can, enable it for locally managed Windows servers, SharePoint farms, and supported NetApp and EMC remotely managed hosts. It is used to collect data on identities, reads, writes, creates, deletes, renames and security changes on securable objects. This information is required for several report types, including the Resource Activity report.

Parameter	Description
Granularity	(Optional) Specify how often (in minutes) you would like to synchronize and aggregate the data. That is, this is the amount of time the agent is to record new activity before sending results to the Data Governance server. The value entered will be changed to a valid aggregation interval, as follows: • Values less than 10 minutes will be set to 5 minutes. • Values between 10 minutes and 2 hours will be set to 1 hour. • Values between 2 hours and 15 hours will be set to 8 hours. • Values greater than 15 hours will be set to 1 day. NOTE: Identical activity generated during this time will be recorded as one activity.
ExcludedTrusteesImportFile	(Optional) Specify the path to a file containing a list of accounts to be excluded from the index scans. This parameter only applies to managed hosts with resource activity enabled.
ExcludedFileTypesImportFile	(Optional) Specify the path to a file containing a list of file types to be excluded from the index scans. This parameter only applies to managed hosts with resource activity enabled.
ExcludedFoldersImportFile	(Optional) Specify the path to a file containing a list of the folders on the computer (paths) to be excluded from the index scans. This parameter only applies to managed hosts with resource activity enabled.
AgentHostName	For remote managed hosts, provide the name of the computer where the scanning agent will reside.
SelectedDataRoots	Specify one or more NTFS directories (or a point in your SharePoint farm hierarchy) to

Parameter	Description
	be scanned by the agent. By default, everything under a selected data roots (paths) is scanned. For remote managed hosts and SharePoint hosts, define the paths to be scanned. For local managed hosts, the agent performs a full scan of the computer by default; however, you can optionally specify the paths to be scanned by the agent. Once configured, only those managed paths are scanned.
ScheduleType	Specifies the time and frequency with which the agent scans the target computer. Valid values are: • DaysOfWeek: Use to specify a daily scan schedule. If you specify this value, you must also specify the RunOnDays and ScheduledTime parameters. • Interval: Use to scan the target computer on an hourly interval instead of a daily schedule. If you specify this value, you must also specify the ScanInterval parameter. • RunOnce: Use to scan the target computer only one time. This parameter is required for remotely scanned managed hosts.
RunOnDays	If the ScheduleType is set to "DaysOfWeek", specify the days you would like the agent to scan the managed host. The syntax is DayOne for Sunday, DayTwo for Monday, etc. For example, to set a scan schedule for Monday, Wednesday and Friday, you would specify ScheduledDays DayTwo,DayFour,DaySix. For remote managed hosts, optionally specify this parameter to define the days of the week to be included in the scan schedule.

Parameter	Description
	If this parameter is not specified, all days of the week are included by default.
ScheduledTime	If the ScheduleType is set to "DaysOfWeek", specify the time of day when the scan is scheduled to start. The syntax is, hh:mm:ss. For example, to start a scan at 4 a.m., specify -ScheduledTime 4:00:00; for 6 p.m., specify -ScheduledTime 18:00:00. For remote managed hosts, optionally specify this parameter to define the time of day when the scan is scheduled to start. If this parameter is not specified, the default start time is 2:00:00 AM.
ScanInterval	If the ScheduleType is set to "Interval", specify the interval (in hours) at which the agent will scan the managed host. For example, to scan every 4 hours, specify -ScanInterval 4. If this parameter is not specified, the default is 24 hours (or 1 day).
ServiceAccountId	If deploying a remotely managed host, you must supply the GUID of the service account that the agent will use to access the remote managed hosts files. Run the Get-QServiceAccounts cmdlet to get a list of service accounts registered with Data Governance Edition and their IDs.
EnableRemoteFileSystemChangeWatching	(Optional) Specify this parameter if you want to collect activity for real-time security updates for the scanned managed host. NOTE: Real-time security updates in the context of Data Governance Edition refers to the monitoring of changes to the file system caused by create, delete, and rename operations, as well as DACL, SACL and Owner changed, in order to maintain the security index. These real-

Parameter	Description
	time security updates are not monitored by default.
OverrideScanScheduleOnStartup	(Optional) Set this flag when you want the agent to do a full scan immediately when the agent is added, or perform a rescan when the agent service is restarted. Valid values are: - 1 or $true: Perform scan when agent is started or restarted. (Default for local managed hosts). If the parameter is specified without a value, set to $true and perform a full scan when agent is started or restarted. - 0 or $false: Do not perform scan when agent is started or restarted. (Default for remote managed hosts.) If the parameter is not specified, set to $false and do not perform a full scan when agent is started or restarted. For example, to override the scan schedule when an agent is started or restarted: -OverrideScanScheduleOnStartup 1
HostType	(Optional) Specify the type of computer the agent will be monitoring. Valid values include: - WindowsServer (Default) - OnTapDevice - CelerraDevice - WindowsCluster - SharePointFarm - GenericHostType - DistributedFileSystemRoot - IsilonDevice - IsilonNfsDevice - OnTapNfsDevice

Parameter	Description
	- OnTapClusterNfsDevice - OnTapClusterCifsDevice If this parameter is not specified, WindowsServer is the default host type.
DataRootType	(Optional) Specify the type of data root. Valid values include: - Share - Folder If this parameter is not specified, defaults to Folder.
IsManagedResourceHost	(Optional) Specify this parameter if you want this managed host to be used to host managed resources (for example, file shares created through the IT Shop self-service request functionality). - $false: (Default) Can not host a managed resource - $true: Can host a managed resource
IgnoreFiles	(Optional) Specify if you want the scanner to include files that have explicit permissions set. If this switch parameter is not present, the managed host scanner will ignore files. This flag is purely for scanning optimization.

Examples:

Table 166: Examples

Example	Description
Add-QManagedHostByAccountName -HostAccountName QAMAUTODC -Keyword QAMAUTO3 -SelectedDataRoot "\\qamautodc\C$\autoroot	Adds a local managed host to the computer "QAMAUTODC", with a keyword of QAMAUTO3. The data root is set to \\qamautocd\C$\autoroot, which means that the agent will only scan this folder and its subfolders on the managed host.
Add-QManagedHostByAccountName -HostAccountNames QAMAUTODC -Keyword	Deploys a remotely scanned managed host, with the agent being hosted on

Example	Description
QAMAUTO -SelectedDataRoot "\\qamautodc\C$\autoroot" -AgentHostName QAMAUTOMEM1 -ServiceAccountId b0a0e218-55c1-41d7-9585-bf7578ad1130 -ScheduleType Interval -ScanInterval 1 -EnableRemoteFileSystemChangeWatching OverrideScanScheduleOnStartup	"QAMAUTODC", with a keyword of QAMAUTO. The dataroot is set as "\\qamautodc\C$\autoroot", For remote managed hosts, you must also include a service account ID, because these are the credentials that the type is set as Interval and the scan interval is set as 1. Remote file resource activity collection is enabled as is override scan schedule on startup. IncludeFiles switch is not included, so the default applies; the scanner will ignore files.
Add-QManagedHostByAccountName -HostAccountName QAMAUTODC -Keyword QAMAUTO3 -SelectedDataRoot "\\qamautodc\C$\autoroot" -IsManagedResourceHost $true	Adds a local managed host that supports the creation of managed resources.
Add-QManagedHostByAccountName SharePoint_ConfigVmset6 vmset6 -AgentHostName QAM-SP2010-DJ -ServiceAccountId 0ca68d5f-f392-453c-9c50-1784332fe3c7 -ResourceActivityEnabled -Granularity 480 -ScheduleType Interval -ScanInterval 1 -OverrideScanScheduleOnStartup -HostType "SharePointFarm" -SelectedDataRoots "SharePoint_ConfigVmset6/SharePoint - 80/My Wiki/My Wiki/Documents|sp://titan/0ee296d6-dea5-4f4d -950f-27c06458cad1/57947f70-c2b0-4d76-a8b3-ac54fa5bb4ab/15c4fc23-b986-4937-890c-d387125d3114/My%20Wiki/Documents"	Adds a SharePoint managed host with one managed path with resource activity enabled.

Clear-QResourceActivity

Clears the resource activity for a given managed host. This enables you to remove activity data from the Data Governance Resource Activity database on demand when it is no longer required.

NOTE: Once activity data is cleared from the database, it cannot be recovered.

Syntax:

```
Clear-QResourceActivity [-ResourceNodeId] <Int32> [<CommonParameters>]
```

Table 167: Parameters

Parameter	Description
ResourceNodeId	Specify the resource node ID of the managed host for which resource activity is to be cleared. This ID is used to link the managed host back to the activity database. Run the Get-QManagedHosts cmdlet to retrieve a list of managed hosts and associated IDs.

Examples:

Table 168: Examples

Example	Description
Clear-QResourceActivity -ResourceNodeId 21	Clears the resource activity from the database for the specified managed host.

Get-QHostsForTrustee

Returns a selected user or group's access on all managed hosts in your environment.

Syntax:

```
Get-QHostsForTrustee [-TrusteeSid] <String> [-IncludeIndirectAccess]
[<SwitchParameter>]] [<CommonParameters>]
```

Table 169: Parameters

Parameter	Description
TrusteeSid	Specify the security identifier (SID) of the account (trustee) whose access you are interested in.
IncludeIndirectAccess	(Optional) Specify this parameter if you want to include indirect access in the results. If this parameter is not specified, the results only includes the managed hosts where the specified account has direct access.

Examples:

Table 170: Examples

Example	Description
Get-QHostsForTrustee -TrusteeSid S-1-5-21-3765505745-248418262-535198764-500	Returns a list of the managed hosts where the specified account has direct access.

Details retrieved:

Table 171: Details retrieved

Detail	Description
HostName	The name of the host to which the account has access.
HostDomainName	The full domain name of the domain to which the managed host computer belongs.
ManagedHostId	The value (GUID) assigned to the managed host computer.
ResourceType	The type of resource to which the account has access.
ViaAccount	For indirect access, the name of the account through which access is being granted.

Get-QManagedHosts

Retrieves a list of managed hosts currently registered with the Data Governance server.

Syntax:

```
Get-QManagedHosts [-HostName [<String>]] [-ManagedHostId [<String>]]
[<CommonParameters>]
```

Table 172: Parameters

Parameter	Description
HostName	(Optional) Specify the pre-Windows 2000 name for the host to be retrieved.
ManagedHostId	(Optional) Specify the ID (GUID format) of the managed host to be retrieved. Run this cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.

Examples:

Table 173: Examples

Example	Description
Get-QManagedHosts	Retrieves a list of all the managed hosts for a given Data Governance Edition deployment.

Example	Description
Get-QManagedHosts -HostName QAMAUTOMEM2	Retrieves the details for the selected managed host.

Details retrieved:

Table 174: Details retrieved

Detail	Description (Associated key or property in QAMManagedHosts table)
Agents	The name and ID (GUID) of agents installed on the managed host. Agents is an array that can be expanded to display the following details about each agent: • Id • ManagedHostId • Management • AgentComputer • AgentComputerDnsName • AgentComputerActiveDirectorySid • AgentComputerManagedDomainId • AgentDetails • UserNotes • PublicKey • ServiceAccountId • IsPrimaryAgent • ConfigurationSettings - this is an array that can be expanded to display the individual configuration settings for the agent. • ScannerStates • LastDugUpdateTimestamp • BelongsToAnotherDeployment
ManagedHostId	The value (GUID) assigned to the managed host computer (ManagedHostId).
ManagedHostSid	The security identifier (SID) assigned to the managed host computer (ManagedHostSid).
ComputerSamSid	Deprecated.

Detail	Description (Associated key or property in QAMManagedHosts table)
ManagedDomainId	The value (GUID) assigned to the managed domain in which the managed host belongs (ManagedDomainId).
HostName	The name of the host (HostName).
DfsRoot	For DFS managed hosts, the value (GUID) assigned to the dfs root to be scanned (DfsRoot).
SamAccountName	The login name for the managed host computer (SAMAccountName).
HostDnsName	The full DNS name of the managed host computer (HostDnsName).
HostDomainName	The full domain name of the domain to which the managed host computer belongs (HostDomainName).
SiteName	If available, the name of the site to which the managed host belongs.
HostType	The physical configuration of the host (HostType).
Management	Indicates whether the host is managed by a local or remote agent (Management): • Local • Remote
Features	The features that a given managed host supports and will allow, such as SecurityIndex and ResourceManagement.
Status	The status of the managed host, based on all the agents monitoring the host.
Internal Status	The status of the managed host, based on all the agents monitoring the host.
ResourceNodeId	The ID used to link the managed host back to the activity database (ResourceNodeId). NOTE: The ResourceNodeId is used in the Clear-QResourceActivity cmdlet.
Keywords	Optional keywords entered when the managed host was added to Data Governance Edition (Keywords).

Detail	Description (Associated key or property in QAMManagedHosts table)
HostContainerId	Deprecated.
SharePointFarmId	For SharePoint managed hosts, the value (GUID) assigned to the SharePoint farm to be scanned (SharePointFarmId).
SharePointFarmObjectGuid	For SharePoint managed hosts, the value (GUID) assigned to the SharePoint object to be scanned (SharePointFarmObjectGuid).
IsManagedResourceHost	Indicates whether this managed host can be used to host managed resources (for example, file shares created through the IT Shop self-service request functionality): - False: Can not host a managed resource. - True: Can host a managed resource.
ApiUserName	The user account used to connect to the target NAS storage device. Only applies to NFS managed hosts and NetApp OnTap Cluster Mode CIFS managed hosts.
ApiPortNumber	The destination port used for communication between the agent and the target NAS storage device. Only applies to NFS managed hosts and NetApp OnTap Cluster Mode CIFS managed hosts.
ResourceActivityTrackingSupported	Indicates whether resource activity collection is enabled.
IsNfsHost	Indicates whether this is an NFS managed host.
IsEmcHost	Indicates whether this is an EMC managed host.
IsNetAppHost	Indicates whether this is a NetApp managed host.

Remove-QManagedHost

Removes a managed host from the list of registered managed hosts.

> NOTE: When unregistered, any agent instances associated with the managed host are also removed. If a computer no longer hosts any agent instances, the Data Governance agent software is also removed.

Syntax:

```
Remove-QManagedHost [-ManagedHostIds] <String[]> [[-DeleteDuGFirst]
[<SwitchParameter>]] [[-SkipAgentUninstall] [<SwitchParameter>]]
[<CommonParameters>]
```

Table 175: Parameters

Parameter	Description
ManagedHostIds	Specify one or managed hosts to be deleted. If you specify multiple managed host ids, separate then with commas. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
DeleteDuGFirst	(Optional) Specify this parameter if you want to remove the entry in the QAMDuG table prior to removing the specified managed hosts.
SkipAgentUninstall	(Optional) Specify this parameter if you want to force the removal of the managed host from the One Identity Manager database, without uninstalling the managed host's agent. If this parameter is not specified, the agent will be removed as part of the host removal process.

Examples:

Table 176: Examples

Example	Description
Remove-QManagedHost -ManagedHostIds A293B96E-9620-4879-8FC7-FB3393E72768	Removes a single managed host from the Data Governance Edition deployment.
Remove-QManagedHost -ManagedHostIds 29F1D9AD-C87A-4F82-830C-0A7CD1088D84,E4A0B8B8-F021-4509-9648-B4C313E542C0	Removes two managed hosts from the Data Governance Edition deployment.

Set-QManagedHostProperties

Changes the properties of a managed host.

Syntax:

```
Set-QManagedHostProperties [-ManagedHostId] <String> [[-Keyword] [<String>]]
[[-ResourceActivityEnabled] [<Boolean>]] [[-Granularity] [<Int32>]] [[-
ScheduleType]
[QAM.Common.Interfaces.ScheduleConfiguration+ScanScheduleTupe>]] [[-
ScheduledDays] [<Int32>]] [[-ScheduledTime] [<TimeSpan>]] [[-ScanInterval]
[<TimeSpan>]] [[-EnableRemoteFileSystemChangeWatching] [<Boolean>]] [[-
PerformImmediateScanOnWatchError] [<Boolean>]] [[-
OverrideScanScheduleOnStartup] [<Boolean>]] [[-SupressHostProcess]
[<SwitchParameter>]] [-IsManagedResourceHost [<Boolean>]]
[<CommonParameters>]
```

Table 177: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host whose properties are to be updated.
Keyword	(Optional) Specify a keyword which can then be displayed and used to group your managed host on the **Managed hosts** view in the Manager.
ResourceActivityEnabled	(Optional) Set this flag to enable resource activity collection. For example: -ResourceActivityEnabled 1
Granularity	(Optional) Specify how often (in minutes) you would like to synchronize and aggregate the data. That is, this is the amount of time the agent is to record new activity before sending results to the Data Governance server. The value entered will be changed to a valid aggregation interval, as follows: • Values less than 10 minutes will be set to 5 minutes. • Values between 10 minutes and 2 hours will be set to 1 hour. • Values between 2 hours and 15 hours will be set to 8 hours. • Values greater than 15 hours will be set to 1 day. NOTE: Identical activity generated during this time will be recorded as one activity.
ScheduleType	Specifies the time and frequency with which the agent scans the target computer. Valid values are:

Parameter	Description
	- DayOfWeek: Use to specify a daily scan schedule. If you specify this value, you must also specify the ScheduledDays and ScheduledTime parameters. - Interval: Use to scan the target computer on an hourly interval instead of a daily schedule. If you specify this value, you must also specify the ScanInterval parameter. This parameter is required for remotely scanned managed hosts.
ScheduledDays	If the ScheduleType is set to "DayOfWeek", specify the days you would like the agent to scan the managed host. The syntax is DayOne for Sunday, DayTwo for Monday, etc. For example, to set a scan schedule for Monday, Wednesday and Friday, you would specify ScheduledDays DayTwo,DayFour,DaySix.
ScheduledTime	If the ScheduleType is set to "DayOfWeek", specify the time of day when the scan is scheduled to start. The syntax is, hh:mm:ss. For example, to start a scan at 4 a.m., specify -ScheduledTime 4:00:00; for 6 p.m., specify -ScheduledTime 18:00:00.
ScanInterval	If the ScheduleType is set to "Interval", specify the interval (in hours) at which the agent will scan the managed host. For example, to scan every 4 hours, specify -ScanInterval 4.
EnableRemoteFileSystemChangeWatching	(Optional) Set this flag to enable change watching for remotely scanned managed hosts. For example: -EnableRemoteFileSystemChangeWatching 1
PerformImmediateScanOnWatchError	(Optional) Set this flag to perform a full scan when the watcher encounters an error. For example:

Parameter	Description
	-PerformImmediateScanOnWatchError 1
OverrideScanScheduleOnStartup	(Optional) Set this flag for a remote managed host when you want the agent to do a full scan when the agent is started or restarted. For example: -OverrideScanScheduleOnStartup 1
SupressHostProcess	(Optional) Specify this parameter to stop the cmdlet from processing the managed host. That is, you can change a managed host's properties without actually triggering the server to use them right away.
SelectedDataRoots	Specify the managed paths where the agent should start scanning. A managed path is the root of an NTFS directory tree to be scanned by an agent, or a point in your SharePoint farm hierarchy below which everything is scanned. The agent monitors the specified managed paths for changes to security settings to maintain the security index. In addition, if resource activity collection is enabled, the agent collects resource activity for these same managed paths. For local managed hosts, all NTFS drives are scanned and monitored by default; However, you can optionally specify the managed paths to be scanned by the agent. When paths are added to this list, only the specified paths are scanned and monitored. For remote managed hosts, you must specify the paths to be managed in order for scanning to occur. So if you do not specify any managed paths using the parameter, no scanning will occur for the target managed host. For SharePoint managed hosts, you must specify the paths to be managed in order for scanning to occur. When you select a point in your SharePoint hierarchy as a managed path, new items added below that point are automatically scanned.

Parameter	Description
IsManagedResourceHost	(Optional) Specify this parameter to change the flag that indicates whether the managed host can be used to host a managed resource (for example, file shares created through the IT Shop self-service request functionality). Valid values are: • $false: Can not be used to host a managed resource (default) • $true: Can be used to host a managed resource

Examples:

Table 178: Examples

Example	Description
Set-QManagedHostProperties -ManagedHostId 97dbedb3-6b02-4dbf-afe2-70d6bf51185a -ResourceActivityEnabled 1	Enables resource activity tracking on the specified managed host.
Set-QManagedHostProperties -ManagedHostId d589359a-8c51-4de0-8dcf-6b463793b0bf -SelectedDataRoots "\\2K8R2DJSQL\C$\Test Data"	Defines a single data root.
Set-QManagedHostProperties -ManagedHostId 97dbedb3-6b02-4dbf-afe2-70d6bf51185a -IsManagedResourceHost $true	Enables managed resources for the managed host.

Set-QManagedHostUpdated

Informs the Data Governance server that the managed host state should be updated.

Syntax:

```
Set-QManagedHostUpdated [-ManagedHostId] <String> [<CommonParameters>]
```

Table 179: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host whose state should be updated.

Examples:

Table 180: Examples

Example	Description
Set-QManagedHostUpdated -ManagedHostId 6834E1A6-B6C5-4508-867A-1E85B7B81578	Updates the managed host specified by the given managed host id.

Trigger-QDfsSync

By default the Data Governance server synchronizes the DFS structure into the One Identity Manager database every 24 hours. Use this cmdlet to force a DFS synchronization of a DFS managed host, making the DFS path immediately available within the Resource browser.

Syntax:

```
Trigger-QDfsSync [-ManagedHostId] <String> [<CommonParameters>]
```

Table 181: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the DFS managed host to be synchronized. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs. TIP: To synchronize all DFS managed hosts in your Data Governance Edition deployment, set the -ManagedHostId to All.

Examples:

Table 182: Examples

Example	Description
Trigger-QDfsSync -ManagedHostId	Forces a synchronization of the specified

Example	Description
f9568450-7396-47ed-bfed-e1377946c2af	DFS managed host.
Trigger-QDfsSync -ManagedHostId All	Forces a synchronization of all DFS managed hosts.

Account access management

As people join, depart, and move through your organization, you need to change their data access. With Data Governance Edition, you can validate that users and groups have been granted access to all the resources they need, ensure that they do not have access to excess resources, and manage their access when problems arise.

The following commands are available to you to manage account access. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 183: Account access management commands

Use this command	If you want to
Get-QAccountAccess	View where users and groups have access on a managed host. For more information, see Get-QAccountAccess on page 172. NOTE: This PowerShell cmdlet does not support Cloud managed hosts.
Get-QAccountAccessOnHosts	View the resource access for a given account (Domain\SAMAccountName) across all available hosts. For more information, see Get-QAccountAccessOnHosts on page 174. NOTE: This PowerShell cmdlet does not support Cloud managed hosts.
Get-QAccountActivity	View the activity associated with a user on a managed host. For more information, see Get-QAccountActivity on page 176. NOTE: This PowerShell cmdlet does not support Cloud managed hosts.
Get-QAccountAliases	View the group membership for a specified account. For example, if one of these groups (aliases) has access to a resource, the original account also has this access. For more information, see Get-QAccountAliases on page 178.

Use this command	If you want to
Get-QAccountsForHost	View all account access for a specific managed host. For more information, see Get-QAccountsForHost on page 179.
Get-QADAccount	View the Active Directory objects from the One Identity Manager and QAM (Data Governance Edition) tables: ADSAccount, ADSGroup, ADSOtherSID, QAMLocalUser and QAMLocalGroup. For more information, see Get-QADAccount on page 180.
Get-QGroupMembers	View all the members of a group, including members of child groups. Because user and group access may be the result of several layers of nested groups, this helps you to assess how a specific account has gained access to a resource. For more information, see Get-QGroupMembers on page 181.
Get-QIndexedTrustees	View all of the entries from the QAMTrustee table who are also listed within the QAMSecurityIndex table, denoting an indexed trustee. For more information, see Get-QIndexedTrustees on page 183.

Get-QAccountAccess

Returns where users and groups have access on a managed host.

Syntax:

```
Get-QAcccountAccess [-ManagedHostId] <String> [-TargetType]
<QAM.Client.PowerShell.TargetType> [-TargetId] <String> [-ResType]
<QAM.Client.PowerShell.QueryResourceType> [[-AccountOrigin] [<String>] [[-
Direct] [<SwitchParameter>]] [[-Exclusions] [<String[]>]] [[-
DataUnderGovernance] [<SwitchParameter>]] [<CommonParameters>]
```

Table 184: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host whose access you are interested in. Run the Get-QManagedHosts command to retrieve a list of managed hosts and their IDs.
TargetType	Specify one of the following types for the target object:

Parameter	Description
	• Account • Identity
TargetId	Specify the ObjectSid for the account or identity.
ResType	Specify the type of resource to be queried. Valid values are: • CloudFiles • CloudFolders • Files • Folders • Shares • LocalOSRights • AdminRights • ServiceIdentities • SharePointResources • SharePointFarmAdminRights • SharePointWebAppPolicies • SharePointSiteCollectionAdminRights
AccountOrigin	(Optional) Specify the origin of the trustee SID specified in the query. Enter the DNS name of the reference domain or computer for the SID. If this parameter is not specified, the server will attempt to infer it.
Direct	(Optional) Specify this parameter if you want the query to retrieve only direct access points. If this parameter is not specified, group membership expansion should be taken into account.
Exclusions	(Optional) Specify a list of trustees that are not to be considered for account access via group membership. This means that if the account being considered is a member of one of the excluded trustees, that access will be ignored. The list must be an array of strings in the following format: [domain DNS name:]SID. The domain DNS name portion can be excluded, in which case Data Governance Edition will infer what it can. For built-in accounts, a missing DNS name means that all of the instances of the provided SID must be excluded.
DataUnderGovernance	(Optional) Specify this parameter if you want to include only

Parameter	Description
	governed resources in your query. If this parameter is not specified, the query will include all resources.

Examples:

Table 185: Examples

Example	Description
Get-QAccountAccess -ManagedHostId 72eed1b9-bf06-4bb9-9ac4-1886daafc514 -TargetId 6a894591-f707-41e5-a187-6b379d07c043 -ResType Folders -AccountOrigin xdomain.local -TargetType Employee -Direct $true	Looks at a managed host with id 72eed1b9-bf06-4bb9-9ac4-1886daafc514. The account or trustee in question has a SID of 6a894591-f707-41e5-a187-6b379d07c043, its type is Identity and the resource type is folders.

Details retrieved:

Table 186: Details retrieved

Detail	Description
RightType	The access right type.
ItemResourceType	The resource type.
ResourceURI	The URI of the resource to which the trustee has access.
TrusteeDisplayName	The display name of the trustee.
TrusteeSid	The SID assigned to the account (trustee).
HostName	The host where the resource resides.
Rights	The specific access rights assigned.
AppliesTo	What the rights apply to.
Inheritance	The type of inheritance.

Get-QAccountAccessOnHosts

For a given account (Domain\SAMAccountName), this cmdlet retrieves the account's resource access across all available hosts.

NOTE: This PowerShell cmdlet does not support Cloud managed hosts.

Syntax:

```
Get-QAccountAccessOnHosts [-AccountName] <String> [-AccountDomain] <String>
[-ManagedHostList [<String>]] [-UriFilterPattern [<String>]] [-DirectOnly [<Switch
Parameter>]] [-ResourceTypes [<String>]] [-OutputDirectory [<String>]] [-
VerboseLogging [<Switch Parameter>]] [<CommonParameters>]
```

Table 187: Parameters

Parameter	Description
AccountName	Specify the name of the account to perform the access report on.
AccountDomain	Specify the name of the domain to perform the access report on.
ManagedHostList	(Optional) Specify the managed hosts to be included in the report. If this parameter is not specified, all managed hosts are included.
UriFilterPattern	(Optional) Specify a string to limit the report to only include resources whose URI contains the given text string.
DirectOnly	(Optional) Specify this parameter to exclude indirect access to a resource from the results.
ResourceTypes	(Optional) Specify the types of resources to be included in the report. Valid resource types are: • Files • Folders • Shares • LocalOSRights • AdminRights • SharePoint (includes all of other SharePoint resource types) • SharePointResourceItems • SharePointFarmAdminRights • SharePointWebAppPolicies • SharePointSiteCollectionAdminRights If this parameter is not specified, all resource types are included.
OutputDirectory	(Optional) Specify an absolute path to a directory where the results are to be saved. If the directory does not exist, it will be created. If this parameter is not specified, the results are only written to the PowerShell output stream.
VerboseLogging	(Optional) Specify this parameter to turn on verbose logging.

Examples:

Table 188: Examples

Example	Description
Get-QAccountAccessOnHosts -AccountName Administrator -AccountDomain MyDomain -ResourceTypes @("SharePoint", "Folders") -OutputDirectory "C:\log.txt" -VerboseLogging	Retrieves all SharePoint and folder access for account "Administrator" in domain "MyDomain". Verbose logging is enabled and the results will be saved in C:\log.txt.

Details retrieved:

Table 189: Details retrieved

Detail	Description
RightType	The access right type.
ItemResourceType	The resource type.
ResourceURI	The URI of the resource to which the trustee has access.
TrusteeDisplayName	The display name of the trustee.
TrusteeSid	The SID assigned to the account (trustee).
HostName	The host where the resource resides.
Rights	The specific access rights assigned.
AppliesTo	What the rights apply to.
Inheritance	The type of inheritance.

Get-QAccountActivity

Retrieves the activity associated with a user on the specified managed host.

NOTE: This PowerShell cmdlet does not support Cloud managed hosts.

Syntax:

```
Get-QAccountActivity [-Trustees] <String[]> [-ManagedHostId] <String> [[-
Extensions] [<String[]>]] [[-StartTime] [<DateTime>]] [[-EndTime]
[<DateTime>]] [<CommonParameters>]
```

Table 190: Parameters

Parameter	Description
Trustees	The security identifier (SID) of the account whose activity you are interested in.
ManagedHostId	The ID (GUID format) of the managed host you would like to see activity for. Run the Get-QManagedHosts command to retrieve a list of managed hosts and their associated IDs.
Extensions	(Optional) Specify the extensions of the file types to be excluded from the query.
StartTime	(Optional) Specify the start date and time (UTC) if you only want to see activity for a time span. Specify the start time in the following format: "23/01/2016 10:36.30 PM"
EndTime	(Optional) Specify the end date and time (UTC) if you only want to see activity for a time span. Specify the end time in the following format: "23/01/2016 10:37.30 PM"

Examples:

Table 191: Examples

Example	Description
Get-QAccountActivity S-1-5-21-3263556741-3296809600-1972185209-1104 3d7e4bb0-e9e2-4d98-b948-21ac7ba1eca6	Returns all the activity for the specified account on the managed host with Id 3d7e4bb0-e9e2-4d98-b948-21ac7ba1eca6.

Details retrieved:

Table 192: Details retrieved

Detail	Description
NodeId	The ID used to link the activity database to the QAMNode table. (AuditNodeId in QAMNode table.)
ManagedHostId	The value (GUID format) assigned to the managed host where the resource is located.
ManagedHostName	The name of the host where the resource is located.
ResourceId	The ID assigned to the operation that was performed.

Detail	Description
ParentResourceId	Shows which resource in the activity database is the parent.
ResourcePath	For file system resources, the path of the resource.
SharePointPath	For SharePoint resources, the path of the resource.
TypeResource	The type of resource.
Operation	The type of operation performed against the resource.
StartTime	The start date and time for collecting resource activity. Activity is stored in 'time spans'.
EndTime	The end date and time for collecting resource activity. Activity is stored in 'time spans'.
TrusteeType	The type of account.
TrusteeName	The display name of the trustee that initiated the operation.
TrusteeSid	The security identifier (SID) assigned to the account (trustee) that initiated the operation.
AuditTrusteeId	The ID associated with the account that performed the operation. (UID_QAMTrustee in QAMTrustee table.)
AccessCount	The number of times the operation occurred during the aggregation interval.

Get-QAccountAliases

Returns the account aliases. This can be used to see the group membership for a specific trustee. For example, if one of these groups (aliases) has access to a resource, the original account will also have this same access.

Syntax:

```
Get-QAccountAliases [-AccountSid] <String> [-AccountDomain] <String>
[<CommonParameters>]
```

Table 193: Parameters

Parameter	Description
AccountSid	Specify the security identifier (SID) of the account.
AccountDomain	Specify the name of the domain the account is in.

Examples:

Table 194: Examples

Example	Description
Get-QAccountAliases -AccountSid S-1-5-21-3765505745-248418262-535198764-1133 mydomain.dge.dev.hal.com	Returns the aliases related to the specified account.

Details retrieved:

Table 195: Details retrieved

Detail	Description
Sid	The security identifier (SID) assigned to the account aliases.
DomainDnsName	The DNS name of the domain where the account is located.
TrusteeType	The type of account.

Get-QAccountsForHost

Retrieves all account access for a specific managed host.

Syntax:

```
Get-QAccountsForHost [-ManagedHostId] <String> [<CommonParameters>]
```

Table 196: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host to be queried. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of managed hosts and associated IDs.

Examples:

Table 197: Examples

Example	Description
Get-QAccountsForHost -ManagedHostId 5b3e4a3c-9c7b-4da1-b6bc-db552ee51656	Retrieves a list of the accounts related to the specified managed host.

Details retrieved:

For each account that has access to the given host, the following information is returned.

Table 198: Details retrieved

Detail	Description
TrusteeName	A list of the accounts (trustees) for the managed host.
TrusteeSid	The security identifier (SID) assigned to each account (trustee).
TrusteeType	The type of account. For a list of trustee types, see Trustee types on page 57
AccessibleHosts	Shows all of the hosts that the account has access to. This host list also shows for each account that has access to the specified host, what other hosts they have access to.

Get-QADAccount

Retrieves Active Directory objects from One Identity Manager and QAM tables: ADSAccount, ADSGroup, ADSOtherSID, QAMLocalUser, and QAMLocalGroup.

Syntax:

```
Get-QADAccount [-Name] [<String>]] [-Domain] [<String>]]
[<CommonParameters>]
```

Table 199: Parameters

Parameter	Description
Name	(Optional) Specify the name of the Active Directory object to be retrieved. If this parameter is not specified, all Active Directory objects are retrieved.
Domain	(Optional) Specify the domain to be queried to locate the Active Directory objects. If this parameter is not specified, all domains are included in the query.

Examples:

Table 200: Examples

Example	Description
Get-QADAccount	Retrieves information for all Active Directory objects on all domains in your Data Governance Edition deployment.
Get-QADAccount -Name Administrator -Domain MyDomain	Retrieves Active Directory information for account Administrator in domain MyDomain.

Details retrieved:

Table 201: Details retrieved

Detail	Description
DomainInfo	DomainInfo is an array that can be expanded to display the following information about the domain the account belongs to: • DnsDomainName • NetbiosDomainName • Type
AccountSid	The security identifier (SID) assigned to the Active Directory account.
SamAccountName	If available, the login name for the account.
DistinquishedName	The distinguished name of the Active Directory account.
Name	The display name of the Active Directory account.
AccountType	The type of account.
ErrorMessage	If available, error messages associated with the Active Directory account.

Get-QGroupMembers

Retrieves a list of all the members of a group, including members of child groups. This helps you assess how a specific account has gained access to a resource.

Syntax:

```
Get-QGroupMembers [-GroupSid] <String> [[-Domain] [<String>]]
[<CommonParameters>]
```

Table 202: Parameters

Parameter	Description
GroupSid	Specify the security identifier, in SDDL format, of the group whose membership you are interested in.
Domain	(Optional) Specify the domain containing the group whose membership you are interested in. NOTE: This value will only be used if the domain is valid and multiple instances of this SID exist (well-known SIDs).

Examples:

Table 203: Examples

Example	Description
Get-QGroupMembers -GroupSid S-1-5-500 -Domain vmset6	Gets the group members from the specified domain.

Detailed retrieved:

Table 204: Details retrieved

Detail	Description
ResultList	ResultList is an array that can be expanded to show the following information for the members of the given group: • ID • ParentID • DNPrefix • SamAccountName • SamAccountType • RID • WellKnown • GroupType • ObjectClass • RedundantBranch
IssueList	IssuesList is an array that can be expanded to view any issues encountered.

Get-QIndexedTrustees

Retrieves all of the entries from the QAMTrustees table who are also listed within the QAMSecurityIndex table, denoting an indexed trustee.

Syntax:

```
Get-QIndexedTrustees [-TrusteeName [<String>]] [-Domain [<String>]]
[<CommonParameters>]
```

Table 205: Parameters

Parameter	Description
TrusteeName	(Optional) Specify the name of the trustee to be searched. If this parameter is not specified, all indexed trustees are returned.
Domain	(Optional) Specify the domain of the trustee to be searched. If this parameter is not specified, all domains are queried to locate indexed trustees.

Examples:

Table 206: Examples

Example	Description
Get-QIndexedTrustees -TrusteeName Administrator -Domain MyDomain	Retrieves all indexed accounts from the QAMTrustees table where the account name is Administrator and the domain is MyDomain.

Details retrieved:

Table 207: Details retrieved

Detail	Description
Sid	The security identifier (SID) assigned to the account.
PreWindows2000Name	The logon name (Pre-Windows 2000) of the Active Directory account.
Domain	The name of the domain where the account resides.
TrusteeType	The type of trustee (account).

Resource access management

A key challenge in improving data governance is keeping track of permissions within your environment. To ensure that data is secured in a manner that meets your business needs, you must be able to easily identify who has been given access and manage that access appropriately.

The following commands are available to you to manage resource access. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 208: Resource access management commands

Use this command	If you want to
Export-QResourceAccess	Export the security information on a selected resource. For more information, see Export-QResourceAccess on page 185.
Get-QChildResources	View the resources contained in a specific root on a managed host. You can use this to enumerate the contents of remote folders and shares. In particular, it would be similar to the standard Windows PowerShell Get-ChildItems cmdlet but it functions using the Data Governance server as a proxy, so the client machine does not necessarily need direct access to the target machine. For more information, see Get-QChildResources on page 187. NOTE: This PowerShell cmdlet does not support Cloud managed hosts.
Get-QFileSystemSearchResults	Search an NTFS folder or share for files. Using this command, you can search multiple data roots at once. For more information, see Get-QFileSystemSearchResults on page 188.
Get-QHostResourceActivities	Retrieve a list of the operations, including the resource ID assigned to each operation, performed against a managed host during a given time frame. For more information, see Get-QHostResourceActivities on page 189. NOTE: This PowerShell cmdlet does not support Cloud managed hosts.
Get-QPerceivedOwners	Calculate the perceived owners for a resource. This information can help to determine the true business

Use this command	If you want to
	owners and custodian for data. NOTE: The perceived owner for data is calculated from the resource activity history or security information collected by Data Governance Edition. Activity is collected based on the aggregation time span settings and recorded in the Data Governance Resource Activity database. For more information, see Get-QPerceivedOwners on page 192.
Get-QResourceAccess	Retrieve the security information of selected resources from a specific managed host, and child objects whose security differs from the parent. For more information, see Get-QResourceAccess on page 194.
Get-QResourceActivity	Retrieve the activity associated with a resource. For more information, see Get-QResourceActivity on page 199. NOTE: Resource activity collection (and therefore this cmdlet) is not supported for the following host types: • Windows Cluster/Remote Windows Computer • Generic Host Type • EMC Isilon NFS Device • SharePoint Online • OneDrive for Business
Get-QResourceSecurity	View the security on a given resource in the SSDL format. For more information, see Get-QResourceSecurity on page 201.
Set-QResourceSecurity	Set security on a given resource. NOTE: The existing security descriptor is completely replaced. For more information, see Set-QResourceSecurity on page 203.

Export-QResourceAccess

Exports the security information on a selected resource to a .CSV file.

TIP: This cmdlet is used with the Get-QResourceAccess cmdlet that generates the results to be exported.

Syntax:

```
Export-QResourceAccess [-ResourceAccessResults]
<QAM.Common.Interfaces.ResourceAccessQueryResults> [-OutputPath] <String>
[[-DisplayInheritedSecurity] [<SwitchParameter>]] [[-OptimizeForExcel]
[<SwitchParameter>]] [<CommonParameters>]
```

Table 209: Parameters

Parameter	Description
ResourceAccessResults	Specify the results of a resource access query (Get-QResourceAccess).
OutputPath	Specify the path to the location on disk where the access results is to be written.
DisplayInheritedSecurity	(Optional) Specify this parameter if child objects with security exactly the same as the parent should be shown. • If the parameter is specified without a value, set to $true and show child objects. • If the parameter is not specified, set to $false and do not show child objects.
OptimizeForExcel	(Optional) Specify this parameter if you want to export the output to Microsoft Excel. • If the parameter is specified without a value, set to $true and export to Excel. • If the parameter is not specified, set to $false and do not export to Excel.

Examples:

Table 210: Examples

Example	Description
C:\PS># get host id Get-QManagedHost # get the access for the resource $resourceAccess = Get-QResourceAccess -ManagedHostId 5b3e4a3c-9c7b-4da1-b6bc-db552ee51656 -ResourceType NTFS\Folder -Resources "C:\Test Data"	Exports the results of a resource access query.

Example	Description
# export the results Export-QResourceAccess -ResourceAccessResults $resourceAccess -OutputPath "C:\"	

Get-QChildResources

Retrieves the resources contained in a specify root on a managed host. You can use this information to enumerate the contents of remote folders and shares.

> NOTE: The cmdlet is similar to the standard Windows PowerShell Get-ChildItems cmdlet, but it functions using the Data Governance server as a proxy. Therefore, the client machine does not require direct access to the target machine.

Syntax:

```
Get-QChildResources [-ManagedHostId] <String> [-ResourcePath] <String> [[-
ResType] [QAM.Client.PowerShell.GetChildResourcesCmdlet+QueryResourceType]]
[<CommonParameters>]
```

Table 211: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host to be queried. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
ResourcePath	Specify the path to the root resource.
ResType	(Optional) Specify the type of resource to be located. Available types include: • CloudFiles • CloudFolders • Files • Folders • Shares • LocalOSRights • AdminRights • ServiceIdentities • SharePoint If this parameter is not specified, all resource types are returned.

Examples:

Table 212: Examples

Example	Description
Get-QChildResources -ManagedHostId 5b3e4a3c-9c7b-4da1-b6bc-db552ee51656 -ResourceId "\\2k8rdjsql\Test Data" -ResType Folders	Retrieves a list of the child resources on the specified managed host.

Details retrieved:

Table 213: Details retrieved

Detail	Description
Path	The full path of the child resource.
DuGPath	The path used for data under governance operations. This will always be empty when shown from the cmdlet; however, it is used elsewhere in the application.
ManagedHostId	The value (GUID) assigned to the managed host where the resource is located.
ResourceType	The type of child resource.
Properties	The properties of the child resource (such as name, date last modified, file size). These are the properties you see in the Resource browser.

Get-QFileSystemSearchResults

Search an NTFS folder or share for files. Using this command, you can search multiple data roots at once.

Syntax:

```
Get-QFileSystemSearchResults [-SearchRoots] <String[]> [-SearchTerm] <String>
[[-ItemsRequested] [<Int32>]] [<CommonParameters>]
```

Table 214: Parameters

Parameter	Description
SearchRoots	Specify a string array of NTFS roots to search.

Parameter	Description
SearchTerm	Specify the string that contains the search term. You can use the * wildcard character to search for resources. For example, enter Finance* to return all resources with a name that begins with Finance, *.txt to return all resources that end with .txt, and *Fin* to return all resources that contain "Fin".
ItemsRequested	(Optional) Specify the number of items you would like returned.

Examples:

Table 215: Examples

Example	Description
Get-QFileSystemSearchResults -SearchRoots "\\2K8R2DJSQL\C$\Test Data" -SearchTerm "*.txt"	Finds files with the .txt extension in the specified directory.

Details retrieved:

The following details are returned for each file system resource found in the specified directory that matched the specified search term.

Table 216: Details retrieved

Detail	Description
Path	The full path of the file system resource.
DuGPath	The path used for data under governance operations. This will always be empty when shown from the cmdlet; however, it is used elsewhere in the application.
ManageHostId	The ID (GUID format) of the managed host where the file system resource resides.
ResourceType	The type of resource.
Properties	Properties assigned to the file system resources (such as Attributes, Reserved, FileSize, LastModified). These are the properties you see in the Resource browser.

Get-QHostResourceActivities

Returns a list of the resource IDs associated with operations performed against a managed host during a given time frame.

NOTE: This PowerShell cmdlet does not support Cloud managed hosts.

Syntax:

```
Get-QHostResourceActivities [[-ManagedHostId] [<String>]] [[-StartTime]
[<DateTime>]] [[-EndTime] [DateTime>]] [[-HostType] [<String>]]
[<CommonParameters>]
```

Table 217: Parameters

Parameter	Description
ManagedHostId	(Optional) Specify the ID (GUID format) of the managed host to be retrieved. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of managed hosts and associated IDs.
StartTime	(Optional) Specify the start date and time, which means you will only see activity information from that time forward. Specify the start time in (UTC) form: "23/01/2015 10:36:30 PM"
EndTime	(Optional) Specify the end date and time, which means you will only see activity information before that time. Specify the end time in (UTC) form: "23/01/2015 10:37:30 PM"
HostType	(Optional) Specify the type of host to be included in the results: • WindowsServer • OnTapDevice • CelerraDevice • SharePointFarm • DistributedFileSystemRoot • IsilonDevice • IsilonNfsDevice • OnTapNfsDevice • OnTapClusterNtfsDevice • OnTapClusterCifsDevice When no host type is specified, all host types are included in the results. NOTE: Resource activity collection is not available for Windows Cluster/Remote Windows or Generic host types.

Examples:

Table 218: Examples

Example	Description
Get-QHostResourceActivities	Returns a list of all activity and associated resource IDs for all activity found in the resource activity database.
Get-QHostResourceActivities -ManagedHostId "c0bc3da4-f660-4e18-8b14-a945c7a6be69"	Returns a list of all activity and associated resource IDs for all host types on the specified managed hosts.
Get-QHostResourceActivities -ManagedHostId "c0bc3da4-f660-4e18-8b14-a945c7a6be69" -HostType "WindowsServer"	Returns activity information on the specified managed host, including only operations against Windows Servers.

Details retrieved:

For each operation performed, the following details are returned:

Table 219: Details retrieved

Detail	Description
NodeId	The ID used to link the activity database to the QAMNode table. (AuditNodeId in QAMNode table.)
ManagedHostId	The ID (GUID) of the managed host reporting the operation.
ManagedHostName	The name of the managed host reporting the operation.
ResourceId	The ID assigned to the operation that was performed.
ParentResourceId	Shows which resource in the activity database is the parent.
ResourcePath	For file system resources, the full path of the resource
SharePointPath	For SharePoint resources, the full path of the resource
TypeResource	The type of resource.
Operation	The type of operation that was performed against the resource: • Create • Delete • Read • Rename • Security change • Write

Detail	Description
StartTime	The start date and time for collecting resource activity. Activity is stored in 'time spans'.
EndTime	The end date and time for collecting resource activity. Activity is stored in 'time spans'.
TrusteeType	The type of account that initiated the operation.
TrusteeName	The name of the user who initiated the operation.
TrusteeSid	The security identifier (SID) of the user who initiated the operation.
AuditTrusteeId	The ID associated with the account that performed the operation. (UID_QAMTrustee in QAMTrustee table.)
AccessCount	The number of times the operation occurred during the aggregation interval.

Get-QPerceivedOwners

Calculates the perceived owners for a resource. You can use this information to determine the true business owners and custodian for data.

> NOTE: The perceived owner for data is calculated from the resource activity history or security information collected by Data Governance Edition. Activity is collected based on the aggregation time span settings and recorded in the Data Governance Resource Activity database.

Syntax:

```
Get-QPerceivedOwners [-ResourcePath] <String> [-ResourceType]
<QAM.Common.Interfaces.ResourceType> [[-NumberOfOwners] [<Int32>]]
[<CommonParameters>]
```

Table 220: Parameters

Parameter	Description
ResourcePath	Specify the full path to the resource whose perceived ownership information is to be queried. For cloud resources, enter the path using the following format: //HostName/root/{path}
ResourceType	Specify the type of resource being queried. Valid values are: • NTFS\Folder • NTFS\File

Parameter	Description
	- Windows Computer\Share - Windows Computer\Local User Rights - Windows Computer\Operating System Administrative Rights - Data Governance\Application Deployment - Service Identities\Windows Service Identity - SharePoint\ResourceItem - SharePoint\WebApplication - SharePoint\SiteCollection - SharePoint\Site - SharePoint\List - SharePoint\Folder - SharePoint\ListItem - DFS\Link - NFS\Folder - NFS\File - Cloud\Folder
NumberOfOwners	(Optional) Specify the number of potential owners to return.

Examples:

Table 221: Examples

Example	Description
Get-QPerceivedOwners -ResourcePath "\\2K8R2DJSQL\C$\Test Data" -ResourceType NTFS\Folder	Calculates and returns the perceived owners for the specified NTFS resource.
Get-QPerceivedOnwers -ResourcePath "//DGEPROD.ONMICROSOFT.COM (SHAREPOINT)/root/Site Contents/Documents/Doc1" -ResourceType Cloud\Folder	Calculates and returns the perceived owners for the specified cloud resource.

Details retrieved:

Table 222: Details retrieved

Detail	Description
TrusteeName	The name of the account returned as a result of the perceived owner calculations.
TrusteeSid	The security identifier (SID) of the account (trustee).
TrusteeType	The type of account.
TotalOperationWeight	The activity weight assigned to the account based on the operations performed during the specified time.
UseCount	The number of times the account accessed the resource during the specified time frame.

Get-QResourceAccess

Retrieves the security information for selected resources from a specific managed host, and child objects whose security differs from the parent. You can retrieve file, folder, share, administrator rights, local operating system rights, and service identity rights.

> TIP: This cmdlet is used with the Export-QResourceAccess cmdlet that exports the saved results.

Syntax:

```
Get-QResourceAccess [-ManagedHostId] <String> [-ResourceType]
<QAM.Client.PowerShell.ResourceAccessQueryResourceType> [[-Resources]
[<String []>]] [-ExcludeSubObjectDeviations [<SwitchParameter>]]
[<CommonParameters>]
```

Table 223: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host that you would like to see access information on. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
ResourceType	Specify the type of rights you would like to see resource access information for. Valid values are: • NTFS\Folder

Parameter	Description
	- NTFS\File - Windows Computer\Share - Windows Computer\Local User Rights - Windows Computer\ Operating System Administrative Rights - Data Governance\Application Deployment - Service Indentities\Windows Service Identity - SharePoint\ResourceItem - SharePoint\WebApplication - SharePoint\SiteCollection - SharePoint\Site - SharePoint\Link - SharePoint\Folder - SharePoint\ListItem - DFS\Link - NFS\Folder - NFS\File - Cloud\Folder
Resources	(Optional) Specify the specific resource you would like to see resource access information for. This parameter only applies to files, folders or shares. To get file and folder security information, specify the network path for remote managed hosts or the local path for local managed hosts. To get share security information, specify the share name only.
ExcludeSubObjectDeviations	(Optional) Specify this parameter to only return the security data for the root objects specified. If this parameter is not specified, the cmdlet returns security information for children below the roots where security differs from the parent.

Examples:

Table 224: Examples

Example	Description

C:\PS>$resourceAccess = Get-QResourceAccess -ManagedHostId 973c7042-c413-45fb-9f52-057c64d4f8aa -ResourceType NTFS\Folder -Resources "C:\Test1","C:\Test2"
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"
	Get file/folder access (local managed host): Retrieves resource access (folder security) for the two folders "C:\Test1" and "C:\Test2" that are located on a local managed host. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet

C:\PS>$resourceAccess = Get-QResourceAccess 973c7042-c413-45fb-9f52-057c64d4f800 -ResourceType NTFS\Folder "\\MachineName\C$\Test1","\\MachineName\C$\Test2"
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"
	Get file/folder access (remote managed host: Retrieves resource access (folder security) for the two folders "\\MachineName\C$\Test1" and "\\MachineName\C$\Test2" that are located on a remote managed host. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet.

C:\PS>$resourceAccess = Get-QResourceAccess 973c7042-c413-45fb-9f52-057c64d4f8aa -ResourceType "Windows Computer\Share" -Resources "ShareName"
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"
	Get share access: Retrieves resource access (share security) for the specified share. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet.

C:\PS>$resourceAccess = Get-QResourceAccess 973c7042-c413-45fb-9f52-057c64d4f800 -ResourceType "Service Identities\Windows Service Identity" -Resources "Dhcp"	Get service identities: Retrieves resource access (entire host) for the security identities on the specified

Example	Description
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"	managed host. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet.

C:\PS>$resourceAccess = Get-QResourceAccess 973c7042-c413-45fb-9f52-057c64d4f800 -ResourceType "Windows Computer\Local User Rights"
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"
	Get local operating system rights: Retrieves resource access (entire host) for the OS rights on the specified managed host. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet.

C:\PS>$resourceAccessInfo = Get-QResourceAccess 973c7042-c413-45fb-9f52-057c64d4f800 -ResourceType "Windows Computer\Operating System Administrative Rights"
C:\PS> Export-QResourceAccess $resourceAccess –OutputPath "C:\ResourceAccessInfo.csv"
	Get administrator rights: Retrieves resource access (entire host) for the admin rights on the specified managed host. The access results are saved to a variable called $resourceAccess which can be exported to a file using the Export-QResourceAccess cmdlet.

Details retrieved:

The most useful information retrieved is the security descriptor details for the specified resource.

Table 225: Details retrieved

Detail	Description
RootResources	RootResources is an array that can be expanded to display the following information: • Id • RootId • Uri • DisplayName • PropertiesString

Detail	Description
	• ResourceSecurityDescriptor • ResourceType • Children
RootResources.ResourceSecurityDescriptor	ResourceSecurityDescriptor under the RootResource parameter is an array that can be expanded to display the following information: • BlockedSecurityInheritance • BlockedAuditingInheritance • InvalidSecurity • NullSecurity • BinarySecurityDescriptor • AceList • ResourceType • SHA1Hash
RootResources.ResourceSecurityDescriptor.AceList	AceList under the ResourceSecurityDescriptor parameter is an array that can be expanded to display the following information for each ACE: • Rights • RightType • Inheritance • AppliesTo • AceTrustee • RawRights • Explicit
RootResources.ResourceSecurityDescriptor.AceList.AceTrustee	AceTrustee under the AceList parameter is an array that can be expanded to display the following information for each

Detail	Description
	account: - Name - Sid - SidType - AuditTrusteeId - UID_QAMTrustee

Get-QResourceActivity

Retrieves the activity associated with a resource. The results provide a granular list of activities recorded over a period of time that can be used to verify proper resource usage and make decisions on modifying access.

> NOTE: Resource activity collection (and therefore, this cmdlet) is not supported for the following host types:
>
> - Windows Cluster/Remote Windows Computer
> - Generic Host Type
> - EMC Isilon NFS Device
> - SharePoint Online
> - OneDrive for Business

Syntax:

```
Get-QResourceActivity [-ManagedHostId] <String> [-Resources] <String[]> [[-StartTime] [<DateTime>]] [[-EndTime] [<DateTime>]] [[-Exclusions] [<String[]>]] [[-ExcludedOperations] [<String[]>]] [<CommonParameters>]
```

Table 226: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host that you would like to see resource activity for. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
Resources	Specify the specific resource you would like to see resource activity for. Specify NTFS resources in the following format: "C:\Share","C:\ADFS"

Parameter	Description
	When specifying multiple resources, separate the resources with a comma.
StartTime	(Optional) Specify the start date and time from which you want to see resource activity. Specify the start time in the following format (UTC): "23/01/2016 10:36:30 PM"
EndTine	(Optional) Specify the end date and time up to which you want to see resource activity. Specify the end time in the following format (UTC): "23/01/2016 11:36:30 PM"
Exclusions	(Optional) Specify the security identifier (SID) of the users to be excluded from the resource activity search. Specify the SIDs to exclude using the following format: domain: S-1-5-21 Example: TSX:S-1-5-21-3263556741-3296809600-1972185209-1104
ExcludedOperations	(Optional) Specify the operations to be excluded from the resource activity search. Valid values are: • Create • Delete • Read • Rename • Security Change • Write When specifying multiple operations, separate the operations with a comma.

Examples:

Table 227: Examples

Example	Description
Get-QResourceActivity "ce21c3ec-3b79-4225-955a-c54cb46790f1" "C:\Share","C:\ADFS"	Retrieves all activity on the specified managed host for the "C:\Share" and "C:\ADFS" folders.

Details retrieved:

Table 228: Details retrieved

Detail	Description
NodeId	The ID used to link the activity database to the QAMNode table. (AuditNodeId in QAMNode table.)
ResourceId	The ID assigned to the operation that was performed.
ParentResourceId	Shows which resource in the activity database is the parent.
ResourcePath	The path of the resource.
ResourceName	The name of resource.
Resource	The type of resource.
Operation	The operation that was performed.
AccessCount	The number of times the operation occurred during the aggregation interval.
StartTime	The start date and time for collecting resource activity. Activity is stored in 'time spans'.
EndTime	The end date and time for collecting resource activity. Activity is stored in 'time spans'.
TrusteeType	The type of account that initiated the operation.
TrusteeName	The name of the account that initiated the operation.
TrusteeSid	The security identifier (SID) assigned to the account that initiated the operation,
AuditTrusteeId	The ID associated with the account that performed the operation. (UID_QAMTrustee in QAMTrustee table.)

Get-QResourceSecurity

Returns the security descriptor for a given resource in the SSDL format.

Syntax:

```
Get-QResourceSecurity [-ResourceUri] <String> [-ResType] <String> [-
DomainDNSName] <String> [[-NoSACL] [<SwitchParameter>]] [[-NoDACL]
[<SwitchParameter>]] [[-NoOwner] [<SwitchParameter>]] [[-NoGroup]
[<SwitchParameter>]] [<CommonParameters>]
```

Table 229: Parameters

Parameter	Description
ResourceUri	Specify the path to the resource for which you want the security descriptor.
ResType	Specify the type of resource in question: - adminrights - localosrights - files - folders - shares
DomainDNSName	Specify the DNS domain name of the domain where the managed host with the resource in question resides.
NoSACL	(Optional) Specify this parameter if you do not want to return the SACL information in the SDDL. If this parameter is not specified, the SACL information will be included.
NoDACL	(Optional) Specify this parameter if you do not want to return the DACL information in the SDDL. If this parameter is not specified, the DACL information will be included.
NoOwner	(Optional) specify this parameter if you do not want to return the Owner information in the SDDL. If this parameter is not specified, the owner information will be included.
NoGroup	(Optional) Specify this parameter if you do not want to return the group information in the SDDL. If this parameter is not specified, the group information will be included.

Examples:

Table 230: Examples

Example	Description
Get-QResourceSecurity -ResourceUri "\\QAMAUTOMem1\c$\autoroot\test_folder" -ResType Folders -DomainDNSName QAMAUTO.QC.HAL.CA.QSFT	Returns the security descriptor for the specified resource on QAMAUTOMem1 in the specified domain.

Set-QResourceSecurity

Sets or updates the security on a given resource to the specified security descriptor.

NOTE: The existing security descriptor is completely replaced.

Syntax:

```
Set-QResourceSecurity [-SDDL] <String> [-ResourceUri] <String> [-ResType]
<String> [-DomainDNSName] <String> [-HostDownLevelName] <String>
[<CommonParameters>]
```

Table 231: Parameters

Parameter	Description
SDDL	Specify the security descriptor (SDDL format) to be set.
ResourceUri	Specify the path to the resource that you want to set the security for.
ResType	Specify the resource type of the resource to have its security descriptor set. Valid values are: • adminrights • localosrights • files • folders • shares
DomainDNSName	Specify the DNS name of the resource's domain.
HostDownLevelName	Specify the downlevel name of the host where the resource resides.

Examples:

Table 232: Examples

Example	Description
Set-QResourceSecurity -SDDL "O:BAG:DUD:AI(A;;FA;;;BA)(A;OICIID;FA;;;BA)(A;OICIID;FA;;;SY)(A;OICIIOID;GA;;;CO)(A;OICIID;0x1200a9;;;BU)(A;CIID;LC;;;BU)(A;CIID;DC;;;BU)S:PAI" -ResourceUri "\\QAMAUTOMem1\c$\autoroot\test_	Sets the security on the specified resource to the specified SDDL on the computer qamautomem1 in the domain qamauto.qc.hal.ca.qsft.

Example	Description
folder" -ResType Folders -DomainDNSName QAMAUTO.QC.HAL.CA.QSFT -HostDownLevelName QAMAUTOMem1	

Governed data management

Governing unstructured data allows you to manage data access, preserve data integrity, and provide content owners with the tools and workflows to manage their own data.

The following commands are available to you to manage governed data. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 233: Governed data management commands

Use this command	If you want to
Get-QDataUnderGovernance	View the data within your organization that has been placed under governance. Data is considered “governed” when it has been explicitly placed under governance or published to the IT Shop. For more information, see Get-QDataUnderGovernance on page 206.
Get-QPerceivedOwnerPoI	View the name of the perceived owner for the specified governed resource. You can use the calculated perceived owners to identify potential business owners for data within your environment. For more information, see Get-QPerceivedOwnerPol on page 209.
Get-QSelfServiceClientConfiguration	View the options that are available for self-service requests within the IT Shop. For more information, see Get-QSelfServiceClientConfiguration on page 211.
Get-QSelfServiceMethodsToSatisfyRequest	View the group membership that is required to satisfy an access request. When identities request access to a resource, an approval workflow is put into action. Before the request for resource access can be

Use this command	If you want to
	granted, the business owner must select a group to which that identity could be added to fulfill their request. For more information, see Get-QSelfServiceMethodsToSatisfyRequest on page 211. NOTE: This PowerShell cmdlet does not support NFS or Cloud resources (since these types of resources cannot be published to the IT Shop).
Remove-QDataUnderGovernance	Remove data from governance. NOTE: Removing a resource from governance, also removes it from the IT Shop. For more information, see Remove-QDataUnderGovernance on page 213.
Set-QBusinessOwner	Set the business owner on a governed resource to establish a custodian for data. The business owner should be an identity who understands the nature of the data and the list of authorized users. Ownership can be established for an individual identity or for all identities in an application role. For more information, see Set-QBusinessOwner on page 213.
Set-QDataUnderGovernance	Place a resource under governance. Once data is "governed", the Data Governance server periodically queries the agent responsible for scanning that data and retrieves detailed security information concerning it and any child data. The data is then placed in the central database to be used by policies and attestations. You can also use this command to set the business owner on governed resources to establish a custodian for data. The business owner should be an identity who understands the nature of the data and the list of authorized users. Ownership can be established for an individual identity or for all identities in an application role.

Use this command	If you want to
	For more information, see Set-QDataUnderGovernance on page 216.
Set-QSelfServiceClientConfiguration	Set the options that are available for self-service requests within the IT Shop. For more information, see Set-QSelfServiceClientConfiguration on page 219.
Trigger-QDataUnderGovernanceCollection	Trigger data collection for governed resources for a given managed host. For more information, see Trigger-QDataUnderGovernanceCollection on page 221.
Upgrade-QDataUnderGovernanceRecords	Upgrade the format of existing governed data in the database after an upgrade from version 6.1.1 or earlier. NOTE: This is a requirement for upgrading to version 6.1.2 or 6.1.3. For more information, see Upgrade-QDataUnderGovernanceRecords on page 221.

Get-QDataUnderGovernance

Retrieves the data within your organization that has been placed under governance.

Syntax:

```
Get-QDataUnderGovernance [[-ResourcePath] [<String>]] [[-ManagedHostId]
[<String>]] [[-MaxResults] [<Int32>]] [<CommonParameters>]
```

Table 234: Parameters

Parameter	Description
ResourcePath	Specify the path to a particular resource under governance. If this parameter is not specified, all resources under governance on the specified managed host are returned. Either the ResourcePath or ManagedHostId parameter must be specified.
ManagedHostId	Specify the ID (GUID format) of the managed host you are interested in.

Parameter	Description
	Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs. Either the ResourcePath or ManagedHostId parameter must be specified.
MaxResults	(Optional) Specify the maximum number of results to be returned. If this parameter is not specified, all results are returned.

Examples:

Table 235: Examples

Example	Description
Get-QDataUnderGovernance -ResourcePath \\QAMAUTOMEM1\C$\AutoRoot\DuG\Folder1	Returns the data under governance object for the resource specified.

Details retrieved:

Table 236: Details retrieved

Detail	Description (Associated key or property in QAMDuG table)
ManagedHostId	Value (GUID) assigned to the managed host computer.
IsForITShop	Indicates if the resource is available for requests through the IT Shop.
DatePublishedToITShop	The date (UTC) when the resource was published to the IT Shop.
IsPublishable	Indicates that the resource is able to be published to the IT Shop.
IsPointOfInterest	Indicates that a point of interest was intentionally placed under governance.
RequiresOwnership	Indicates that the resource requires that an owner be assigned.
DisplayName	Name of the governed resource.
DisplayPath	Path and name of the governed resource.
Description	Descriptive information entered for the governed resource.
FullPath	Full path of the governed resource.

Detail	Description (Associated key or property in QAMDuG table)
FullPathHashSHA1	Hash value over the full path for unique identification.
Justification	The reason for assigning this owner to the resource.
OwnershipSetBy	Name of the account that set the owner.
PlacedUnderGovernanceBy	Name of the account that placed the resource under governance.
RiskIndex	Calculated risk index of all assignments to this data.
ActivityResourceId	The value that relates the roots in this database to data in the Data Governance activity resource database.
DateOwnershipSet	The date (UTC) when the ownership of the resource was set.
UID_QAMDuG	The identifier assigned to the governed resource by Data Governance Edition.
IsStale	Indicates whether the resource was renamed or deleted.
LastEncounteredTime	The time detailed security information was successfully collected.
PersonOwnerKey	If you have assigned an identity as the business owner of this resource, this is the primary key of that identity.
PersonOwnerDisplay	If an identity is assigned as the business owner, the name of that identity.
RoleOwnerKey	If you have assigned a role as the business owner of this resource, this is the primary key of that role.
RoleOwnerDisplay	If an application role is assigned as the business owner, the name of that application role.
ResourceType	The governed data type.
ManagedHostName	The name of the managed host computer.
UseBackingFolderSecurity	Indicates to use the backing folder of a share.
LastPoiCollection	The date (UTC) when the POI was last collected.
LastPoiSubmission	The date (UTC) when the POI was last submitted.
Security	The security used for governance. (SecurityForGovernance)
ClassificationLevelId	If a classification level is assigned, the identifier assigned to the classification level. (UID_QAMClassificationLevelMan Value)

Detail	Description (Associated key or property in QAMDuG table)
ClassificationLevelName	If a classification level is assigned, the name assigned to the classification level. (UID_QAMClassificaitonLevelMan)

Get-QPerceivedOwnerPol

Retrieves the name of the perceived owner for the specified governed resource. You can then use the calculated perceived owners to identify potential business owners for data within your environment.

Syntax:

```
Get-QPerceivedOwnerPoI [-GovernedDataId] <String> [<CommonParameters>]
```

Table 237: Parameters

Parameter	Description
GovernedDataId	Specify the ID (GUID format) of the governed resource whose perceived owner information you want to identify. Run the Get-QDataUnderGovernance cmdlet to retrieve a list of governed resources and their associated IDs (UID_QAMDuG value) for a specific managed host.

Examples:

Table 238: Examples

Example	Description

C:\PS>$resources = Get-Content 'C:\Resources.txt'
foreach($resource in $resources)
{
  try
  {
    $governed = Get-QDataUnderGovernance $resource
    if($governed)
    {
      $perceivedOwner = Get-
	Returns the perceived owner information for a governed resource with the specified id.
	This PowerShell script takes a list of governed resources and returns the perceived owner for each.

Example	Description

```
QPerceivedOwnerPoI $governed.UID_QAMDuG
      $resource += ';'
      $resource += $perceivedOwner.EmployeeId
      Add-Content 'c:\PerceivedOwnerResource.txt' $resource
    }
    else
    {
      $resource += ';'
      $resource += 'Resource Not Governed'
       Add-Content 'c:\PerceivedOwnerResource.txt' $resource
    }
  }
  catch
  {
    Writestatus $_
  }
}
```

Details retrieved:

Table 239: Details retrieved

Detail	Description (Associated key or property in QAMPoIPerceivedOwner table)
EmployeeName	The name of the perceived owner (identity) for the governed resource.
EmployeeId	The value (GUID) assigned to the perceived owner (identity).
TrusteeName	The name of the account that initiated the operation.
TrusteeId	The value (GUID) assigned to the trustee (UID_QAMTrustee).
TrusteeXObjectKey	The value (<Key>) assigned to the account.
TrusteeType	The type of account.

Get-QSelfServiceClientConfiguration

Returns the options available for self-service requests within the IT Shop.

Syntax:

```
Get-QSelfServiceClientConfiguration [<CommonParameters>]
```

Examples:

Table 240: Examples

Example	Description
Get-QSelfServiceClientConfiguration	Returns the self-service client configuration information.

Details retrieved:

Table 241: Details retrieved

Detail	Description
AllowNonPublishedGroups	Indicates whether groups that have not been published to the IT Shop are allowed for self-service access requests.
AllowUnsynchronizedGroups	Indicates whether groups that have not been synchronized with One Identity Manager are allowed for self-service access requests.
MaximumMethodsCount	The maximum number of groups returned from a call to the Get-QSelfServiceMethodsToSatisfyRequest, which returns the groups that satisfy a resource access request.
EnableSelfServiceAccessRequest	Indicates whether self-service access requests are enabled in the IT Shop.

Get-QSelfServiceMethodsToSatisfyRequest

Returns the group membership that satisfies a resource access request. Use this command to simulate the "best fit" calculation to see what groups are returned if you request read or read and write access to a specific resource.

> NOTE: This PowerShell cmdlet does not support NFS or Cloud resources (since these types of resources cannot be published to the IT Shop).

Syntax:

```
Get-QSelfServiceMethodsToSatisfyRequest [-Path] <String> [-DomainName]
<String> [-ActionIdentifier] <String> [[-ClientCulture] [<String>]] [[-
ResourceTypeString] [<String>]] [<CommonParameters>]
```

Table 242: Parameters

Parameter	Description
Path	Specify the path of the resource.
DomainName	Specify the name of the domain where the resource is located.
ActionIdentifier	Specify the type of self-service action: • RequestReadAccess: Use this option if you want read access to items within a folder. • RequestChangeAccess: Use this option if you want read and write access to items within a folder.
ClientCulture	(Optional) Set the client culture.
ResourceTypeString	(Optional) Specify the type of resource for which to request access: • NTFS\Folder • NTFS\File • Windows\Computer\Share • SharePoint\Site • SharePoint\Folder • SharePoint\List • SharePoint\ListItem • SharePoint\ResourceItem

Examples:

Table 243: Examples

Example	Description
Get-QSelfServiceMethodsToSatisfyRequest -Path "\\2K8RDJSQL\CS\Test Data\Email_Addresses.txt" -DomainName VMSET6 -ActionIdentifier "RequestReadAccess" -ResourceTypeString NTFS\File	Returns the groups that satisfy the "RequestReadAccess" request for a NTFS/File.

Remove-QDataUnderGovernance

Removes a resource from governance, and if published to the IT Shop, removes it from the IT Shop.

Syntax:

```
Remove-QDataUnderGovernance [-ResourceUri] <String> [<CommonParameters>]
```

Table 244: Parameters

Parameter	Description
ResourceUri	Specify the Uri of the resource to be removed from governance. Use the following format for files and folders: "\\MACHINE\DRIVELETTER$\PathToResource". TIP: If you are having trouble with SharePoint paths, use the Resource browser (in the Manager) to copy the SharePoint path.

Examples:

Table 245: Examples

Example	Description
Remove-QDataUnderGovernance -ResourceUri "\\2k8r2djsql\C$\Test Data"	Removes an NTFS resource from governance.
Remove-QDataUnderGovernance -ResourceUri "sp://titan/6d338b7c-79cc-4b99-a1d0-47641cc0cebc/42d1bc72-8754-4b7d-8bac-0be07d07e8f2/faa56136-6317-4c31-9e90-649347df4bed/DerekSite/Shared%20Documents/My%20SharePoint%20Doc.txt"	Removes a SharePoint resource from governance.

Set-QBusinessOwner

Set the business owner of a resource under governance.

> NOTE: This command only works for resources that have previously been placed under governance.

Syntax

```
Set-QBusinessOwner [-ManagedHostId] <String> [[-ResourceUri] [<String>]] [[-
SetAllResources] [<Boolean>]] [[-EmployeeName] [<String>]] [[-EmployeeId]
[<String>]] [-OwnerRoleFullPath] [<String>]] [[-OwnerRoleId] [<String>]] [[-
Justificaiton] [<String>]] [<CommonParameters>]
```

Table 246: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host where the governed resource resides.
ResourceUri -OR- SetAllResources	Specify one of the following parameters to specify the governed resource that is to be assigned a business owner: • ResourceUri: Use this parameter to set the business owner for a single governed resource. Enter the resource's full path. For example (to specify a share): -ResourceURI "\\dgefs\a1" For example (to specify a NTFS folder): -ResourceURI "\\dgefs\D$\al" • SetAllResources: Use this parameter to set the same business owner to all governed resources on the specified managed host. Set this value to 1. For example: -SetAllResources 1 NOTE: You must specify one of these parameters to specify the governed resource. Do NOT specify more than one of these parameters or you will receive an error when running the Power-Shell command.
EmployeeName -OR- EmployeeId -OR- OwnerRoleFullPath -OR- OwnerRoleId	Specify one of the following parameters to define the business owner to be assigned: • EmployeeName: Specify the name of the identity to be assigned as the business owner. For example: -EmployeeName "user6 test, user6" • EmployeeId: Specify the ID (GUID format) of the identity to be assigned as the business owner. For example: -EmployeeId 3dd99328-e971-4bcf-989e-9a482871e9e9 • OwnerRoleFullPath: Specify the full path of a One Identity Manager application role if you want all identities in the selected role to be the business owner. For example: -OwnerRoleFullPath "Data Governance\All Business Owner Roles\Finance Owners"

Parameter	Description
	- OwnerRoleId: Specify the ID (GUID format) of a One Identity Manager application role if you want all identities in the selected role to be the business owner. For example: -OwnerRoleId 50b8b7b8-6670-4e35-bd3b-f6f64a281364 NOTE: You must specify one of these parameters to define the business owner. Do NOT specify more than one of these parameters or you will receive an error when running the PowerShell command.
Justification	(Optional) Enter a reason for setting the business owner.

Examples

Table 247: Examples

Example	Description
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -ResourceURI "\\dgefs\a1" -EmployeeName "user6 test, user6"	Sets the business owner for a single resource, using the identity's name.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -ResourceURI "\\dgefs\a1" -EmployeeId 3dd99328-e971-4bcf-989e-9a482871e9e9	Sets the business owner for a single resource, using the identity's ID.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -ResourceURI "\\dgefs\a1" -OwnerRoleId 50b8b7b8-6670-4e35-bd3b-f6f64a281364	Sets the business owner for a single resource, using an application role ID. All identities assigned to this role are considered the business owner.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -ResourceURI "\\dgefs\a1" -OwnerRoleFullPath Data "overnance\All Business Owner Roles\Finance Owners"	Sets the business owner for a single resource, using an application role path. All identities assigned to this role are considered the business owner.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -SetAllResource 1 -EmployeeName "user6 test, user6"	Sets the business owner for all governed resources on the specified managed host, using the identity's name.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -SetAllResource 1 -EmployeeId 3dd99328-e971-4bcf-989e-9a482871e9e9	Sets the business owner for all governed resources on the specified managed host, using the identity's ID.

Example	Description
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -SetAllResource 1 -OwnerRoleId 50b8b7b8-6670-4e35-bd3b-f6f64a281364	Sets the business owner for all governed resources on the specified managed host, using an application role ID. All identities assigned this role are considered the business owner.
Set-QBusinessOwner -ManagedHostId b5552078-9eef-4aa4-99dc-3b556277b3b1 -SetAllResource 1 -OwnerRoleFullPath Data "overnance\All Business Owner Roles\Finance Owners"	Sets the business owner for all governed resources on the specified managed host, using an application role path. All identities assigned this role are considered the business owner.

Set-QDataUnderGovernance

Places a resource under governance.

Syntax:

```
Set-QDataUnderGovernance [-ManagedHostId] <String> [-ResourceType] <String>
[-ResourceUri] <String> [[-DisplayPath] [<String>]] [[-EmployeeName]
[<String>]] [[-EmployeeId] [<String>]] [[OwnerRoleFullPath] [<String>]] [[-
OwnerRoleId] [<String>]] [[-PublishToITShop] [<Boolean>]] [[-
UseBackingFolderSecurity] [<SwitchParameter>]] [[-Reset] [<SwitchParameter>]]
[[-SharePointDisplayPath] [<String>]] [[-ManagedResourceId] [<String>]]
[<CommonParameters>]
```

Table 248: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host where the resource to be placed under governance is located. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.
ResourceType	Specify the type of resource to be placed under governance. Valid values include: • None • FolderType • ShareType • LocalRightType • AdminRightType • QamDeploymentType

Parameter	Description
	• WindowsServiceIdentityType • SharePointResourceItemType • SharePointWebApplicationType • SharePointSiteCollectionType • SharePointSiteType • SharePointListType • SharePointFolderType • SharePointListItemType • SharePointSiteCollectionAdminRightType • SharePointFarmAdminRightType • SharePointWebAppPolicyType • SharePointServiceApplicationPermissionType • SharePointFarmType • DFSLinkType • NFSFolderType • Cloud\Folder
ResourceUri	Specify the Uri for the resource to be placed under governance. For NTFS files and folders, use the form: \\MACHINE\$DRIVELETTER\PathToResource For SharePoint, enter the FARM GUID, Site collection GUID (that is, sp://titan/0ee296d6-dea5-4f4d-950f-27c06458cad1/57947f70-c2b0-4d76-a8b3-ac54fa5bb4ab/203a4c04-4f0e-4d6a-84a7-c2ef0a3f02e3/Dereks%20Site/Shared%20Documents/SharePoint/desktop.ini)
DisplayPath	(Optional) Specify the path of the resource to be displayed in the Manager. This is useful for long paths.
EmployeeName	(Optional) Specify the name of the identity who is set as the business owner of the current governed resource. NOTE: Specify only one of the following parameters: EmployeeName, EmployeeId, OwnerRoleFullPath, or OwnerRoleId.
EmployeeId	(Optional) Specify the ID (GUID format) of the identity who is set as the business owner of the current governed resource. NOTE: Specify only one of the following parameters:

Parameter	Description
	EmployeeName, EmployeeId, OwnerRoleFullPath, or OwnerRoleId.
OwnerRoleFullPath	(Optional) Specify the full path of the application role who is set as the business owner of the current governed resource. NOTE: Specify only one of the following parameters: EmployeeName, EmployeeId, OwnerRoleFullPath, or OwnerRoleId.
OwnerRoleId	(Optional) Specify the ID (GUID format) of the application role who is set as the business owner of the current governed resource. NOTE: Specify only one of the following parameters: EmployeeName, EmployeeId, OwnerRoleFullPath, or OwnerRoleId.
PublishToITShop	(Optional) Specify this parameter to place the resource under governance and add it to the IT Shop. Valid values are: • 0 or $false: Do not publish the resource to the IT Shop. (Default) • 1 or $true: Publish the resource to the IT Shop.
UsingBackingFolderSecurity	(Optional) Specify this parameter to indicate the security for the backing folder is to be used.
Reset	(Optional) Specify this parameter to indicate whether you want to reset the governed resource record if it already exists in the database. Valid values are: • 0 or $false: do not reset the existing QAMDuG entry in the database. (Default) • 1 or $true: reset the existing QAMDuG entry in the database with the new values specified in this cmdlet.
SharePointDisplayPath	(Optional) Specify the readable SharePoint path (that is, SharePoint_ConfigVmset6/SharePoint - 80/Site/Shared Documents/SharePoint) to be displayed in the Manager.
ManagedResourceId	(Optional) Specify this parameter to link the QAMDuG entry to a QAMManagedResource record in the database.

Examples:

Table 249: Examples

Example	Description
Set-QDataUnderGovernance -managedhostid 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A -ResourceType "NTFS\Folder" -ResourceUri \\qamautomem1\C$\autoroot\DUG\Folder2	This example places the resource \\qamautomem1\C$\autoroot\DUG\Folder2 under governance for the managed host identified by 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A.
Set-QDataUnderGovernance -managedhostid 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A -ResourceType "NTFS\Folder" -ResourceUri \\qamautomem1\C$\autoroot\DUG\Folder2 -EmployeeName "Admin, Admin" -Reset $true	This example places the resource \\qamautomem1\C$\autoroot\DUG\Folder2 under governance for the managed host identified by 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A. It also sets the identity whose name is "Admin, Admin" as the business owner of this governed resource. If this governed resource already exists, it would be reset.
Set-QDataUnderGovernance -managedhostid 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A -ResourceType "NTFS\Folder" -ResourceUri \\qamautomem1\C$\autoroot\DUG\Folder2 -OwnerRoleId "81d50b9e-7ab6-43c0-8016-da972e633303" -Reset $true	This example places the resource \\qamautomem1\C$\autoroot\DUG\Folder2 under governance for the managed host identified by 68CD6FFC-BA2C-4F8E-8C34-70D2C1C1995A. It also sets the role whose Id is "81d50b9e-7ab6-43c0-8016-da972e633303" as the business owner of this governed resource. If this governed resource already exists, it would be reset.
Set-QDataUnderGovernance -managedhostid ca990043-8ffc-4eea-8e11-179a1d3505af -ResourceType SharePoint\ListItem -ResourceURI 'sp://titan/0ee296d6-dea5-4f4d-950f-27c06458cad1/57947f70-c2b0-4d76-a8b3-ac54fa5bb 3a4c04-4f0e-4d6a-84a7-c2ef0a3f02e3/Dereks%20Site/Shared%20Documents/SharePoint/desktop.ini' -Displaypath SharePoint_ConfigVmset6/SharePoint - 80/Dereks Site/Dereks Site/Shared Documents/SharePoint/desktop.ini'	This example places the resource sp://titan/0ee296d6-dea5-4f4d-950f-27c06458cad1/57947f70-c2b0-4d76-a8b3-ac54fa5bb4ab/203a4c04-4f0e-4d6a-84a7-c2ef0a3f02e3/Dereks%20Site/Shared%20Documents/SharePoint/desktop.ini under governance for the managed host identified by ca990043-8ffc-4eea-8e11-179a1d3505af.

Set-QSelfServiceClientConfiguration

Sets the options available for self-service requests within the IT Shop.

Syntax:

```
Set-QSelfServiceClientConfiguration [-MaximumMethodsCount] <Int32> [-
EnableSelfServiceAccessRequest] <Boolean> [-AllowNonPublishedGroups]
<Boolean> [-AllowUnsynchronizedGroups] <Boolean> [<CommonParameters>]
```

Table 250: Parameters

Parameter	Description
MaximumMethodsCount	Specify the maximum number of groups that are to be returned from a call to the GetMethodsToSatisfyRequest.
EnableSelfServiceAccessRequest	Specify whether self-service access requests are to be enabled in the IT Shop. Valid values are: • 0: Disable self-service access requests • 1: Enable self-service access requests
AllowNonPublishedGroups	Specify whether groups that have not been published to the IT Shop are to be included in self-service access requests. Valid values are: • 0: Unpublished groups will not be available for self-service access requests in the IT Shop. • 1: Unpublished groups will be available for self-service access requests in the IT Shop.
AllowUnsynchronizedGroups	Specify whether groups that have not been synchronized with One Identity Manager are to be included in self-service requests. Valid values are: • 0: Unsynchronized groups will not be available for self-service access requests in the IT Shop. • 1: Unsynchronized groups will be available for self-service access requests in the IT Shop.

Examples:

Table 251: Examples

Example	Description
Set-QSelfServiceClientConfiguration -MaximumMethodsCount 1 -	Sets the self-service client configuration information:

Example	Description
EnableSelfServiceAccessRequest 1 -AllowNonPublishedGroups 1 -AllowUnsynchronizedGroups 1	• Enabling self-service access requests • Making unpublished groups available for self-service access requests in the IT Shop • Making unsynchronized groups available for self-service access requests in the IT Shop

Trigger-QDataUnderGovernanceCollection

Triggers data collection on the governed resources for a specific managed host.

Syntax:

```
Trigger-QDataUnderGovernanceCollection [-ManagedHostId] <String>
[<CommonParameters>]
```

Table 252: Parameters

Parameter	Description
ManagedHostId	Specify the ID (GUID format) of the managed host where data collection is to take place. Run the Get-QManagedHosts cmdlet without any parameters to retrieve a list of available managed hosts and their IDs.

Examples

Table 253: Examples

Example	Description
Trigger-QDataUnderGovernanceCollection -ManagedHostId d589359a-8c51-4de0-8dcf-6b463793b0bf	Triggers the collection of access information for resources under governance.

Upgrade-QDataUnderGovernanceRecords

Upgrades the format of existing governed data in the database after an upgrade from Version 6.1.1 or earlier.

NOTE: This is a requirement for upgrading to Data Governance Edition Version 6.1.2 or 6.1.3.

Syntax:

```
Upgrade-QDataUnderGovernanceRecords [<CommonParameters>]
```

Classification management

Classification is included in Data Governance Edition, however you should first define the classification levels in Data Governance Edition to match those defined by your company. Once defined, you can use these classification levels to classify governed resources.

The following commands are available to manage the classification levels used in your Data Governance Edition deployment and to assign a classification level to a governed resource. For full parameter details and examples, click a command hyperlink in the table or see the command help, using the **Get-Help** command.

Table 254: Group template management commands

Use this command	If you want to
Add-QClassificationLevel	Define a new classification level for use in your Data Governance Edition deployment. For more information, see Add-QClassificationLevel on page 223.
Get-QClassificationLevelConfiguration	Retrieve details about the classification levels configured in your Data Governance Edition deployment. For more information, see Get-QClassificationLevelConfiguration on page 224.
Get-QDataUnderGovernanceByClassificationLevel	Retrieve a list of governed resources assigned a specific classification level. For more information, see Get-QDataUnderGovernanceByClassificationLevel on page 224.
Remove-QClassificationLevel	Remove a classification level from your Data Governance Edition deployment. For more information, see Remove-QClassificationLevel on page 227.
Set-QClassificationLevel	Update an existing classification level in your Data Governance Edition deployment.

Use this command	If you want to
	For more information, see Set-QClassificationLevel on page 228.
Set-QClassificationLevelOnDug	Assign a classification level to a governed resource. For more information, see Set-QClassificationLevelOnDuG on page 228.

Add-QClassificationLevel

Defines a new classification level for use in your Data Governance Edition deployment.

Syntax:

```
Add-QClassifictionLevel [-Name] <String> [-Description] <String> [[-SortOrder] <Int>] [<CommonParamters>]
```

Table 255: Parameters

Parameter	Description
Name	Specify the name to be associated with the new classification level. The length of the name is limited to 512 characters. Any text is allowed, including spaces and other 'special characters'.
Description	Enter descriptive text to be associated with the new classification level. Any text is allowed, including spaces and other 'special characters'.
SortOrder	(Optional) Specify the display order of the new classification level. The classification levels are displayed in ascending order based on SortOrder. If no SortOrder value is specified, the classification level will appear at the top of the list.

Examples:

Table 256: Examples

Example	Description
Add-QClassificationLevel -Name "Internal Eyes Only" -Description "Intended for internal distribution within the organization." -SortOrder 1	Adds a new "Internal Eyes Only" classification level.

Get-QClassificationLevelConfiguration

Retrieves details about the classification levels configured in your Data Governance Edition deployment.

Syntax:

Get-QClassificationLevelConfiguration [<CommonParameters>]

Examples:

Table 257: Examples

Example	Description
Get-QClassificationLevelConfiguration	Returns details about each classification levels previously configured, including the classification ID.

Details retrieved:

For each classification level configured, this cmdlet returns the following details.

Table 258: Details retrieved

Detail	Description (Associated property in QAMClassificationLevel table)
Id	The identifier assigned to the classification level by Data Governance Edition (UID_QAMClassificationLevel).
Name	The name of the classification level. For example: Critical Handling (Name).
Description	The descriptive text associated with the classification level (Description).
SortOrder	The display order value assigned to the classification level (SortOrder).

Get-QDataUnderGovernanceByClassificationLevel

Retrieves a list of governed resources assigned a specific classification level.

Syntax:

```
Get-QDataUnderGovernanceByClassificationLevel [-ClassificationLevelId] <String>
[<CommonParameters>]
```

Table 259: Parameters

Parameter	Description
ClassificationLevelId	Specify the identifier assigned to the classification level. Run the Get-QClassificationLevelConfiguration cmdlet to retrieve a list of configured classification levels, including their assigned identifiers.

Examples:

Table 260: Examples

Example	Description
Get-QDataUnderGovernanceByClassificationLevel -ClassificationLevelId 51442B53-A9BE-4EE0-8A89-B5D5ED3CF387	Returns a list of the governed resources associated with the specified classification level.

Details retrieved:

Table 261: Details retrieved

Detail	Description (Associated key or property in QAMDuG table)
ManagedHostId	Value (GUID) assigned to the managed host computer.
IsForITShop	Indicates if the resource is available for requests through the IT Shop.
DatePublishedToITShop	The date (UTC) when the resource was published to the IT Shop.
IsPublishable	Indicates that the resource is able to be published to the IT Shop.
IsPointOfInterest	Indicates that a point of interest was intentionally placed under governance.
RequiresOwnership	Indicates that the resource requires that an owner be assigned.
DisplayName	Name of the governed resource.

Detail	Description (Associated key or property in QAMDuG table)
DisplayPath	Path and name of the governed resource.
Description	Descriptive information entered for the governed resource.
FullPath	Full path of the governed resource.
FullPathHashSHA1	Hash value over the full path for unique identification.
Justification	The reason for assigning this owner to the resource.
OwnershipSetBy	Name of the account that set the owner.
PlacedUnderGovernanceBy	Name of the account that placed the resource under governance.
RiskIndex	Calculated risk index of all assignments to this data.
ActivityResourceId	The value that relates the roots in this database to data in the Data Governance activity resource database.
DateOwnershipSet	The date (UTC) when the ownership of the resource was set.
UID_QAMDuG	The identifier assigned to the governed resource by Data Governance Edition.
IsStale	Indicates whether the resource was renamed or deleted.
LastEncounteredTime	The time detailed security information was successfully collected.
PersonOwnerKey	If you have assigned an identity as the business owner of this resource, this is the primary key of that identity.
PersonOwnerDisplay	If an identity is assigned as the business owner, the name of that identity.
RoleOwnerKey	If you have assigned a role as the business owner of this resource, this is the primary key of that role.
RoleOwnerDisplay	If an application role is assigned as the business owner, the name of that application role.
ResourceType	The governed data type.
ManagedHostName	The name of the managed host computer.
UseBackingFolderSecurity	Indicates to use the backing folder of a share.
LastPoiCollection	The date (UTC) when the POI was last collected.
LastPoiSubmission	The date (UTC) when the POI was last submitted.

Detail	Description (Associated key or property in QAMDuG table)
Security	The security used for governance. (SecurityForGovernance)
ClassificationLevelId	If a classification level is assigned, the identifier assigned to the classification level. (UID_QAMClassificationLevelMan Value)
ClassificationLevelName	If a classification level is assigned, the name assigned to the classification level. (UID_QAMClassificationLevelMan)

Remove-QClassificationLevel

Removes an existing classification level from your Data Governance Edition deployment.

Syntax:

```
Remove-QClassificationLevel -[-ID] <String> [<CommonParameters>]
```

Table 262: Parameters

Parameter	Description
ID	Specify the identifier assigned to the classification level to be removed. Run the Get-QClassificationLevelConfiguration cmdlet to retrieve a list of configured classification levels, including their assigned identifiers. NOTE: Deleting a classification level will automatically remove it from all associated governed data. Prior to running this cmdlet, run the Get-QDataUnderGovernanceByClassificationLevel cmdlet to retrieve a list of the resources assigned to the specified classification level.

Examples:

Table 263: Examples

Example	Description
Remove-QClassificationLevel -ID 4E4F22C7-A30A-45C3-808A-C134C132B590	Removes the specified classification level from your Data Governance Edition deployment.

Set-QClassificationLevel

Updates an existing classification level defined for use by the Data Governance Edition deployment.

Syntax:

```
Set-QClassificationLevel [-ID] <String> [[-Name] [<String>]] [[-Description]
[<String>]] [[-SortOrder] [<Int>]] [<CommonParameters>]
```

Table 264: Parameters

Parameter	Description
ID	Specify the identifier assigned to the classification level to be updated. Run the Get-QClassificationLevelConfiguration cmdlet to retrieve a list of configured classification levels, including their assigned identifiers.
Name	Specify to change the name to be associated with the specified classification level. The length of the name is limited to 512 characters. Any text is allowed, including spaces and other special characters.
Description	Specify to change the descriptive text to be associated with the specified classification level. Any text is allowed, including spaces and other special characters.
SortOrder	Specify to change the display order of the selected classification level.

Examples:

Table 265: Examples

Example	Description
Set-QClassificationLevel -ID D7EADC4B-46F1-430A-95C7-1D300A4E6FA3 -Name "Public" -Description "General information created for internal or external sources that can be shared publicly."	Changes the name and description of the specified classification level.

Set-QClassificationLevelOnDuG

Assigns a classification level to a governed resource.

Syntax:

```
Set-QClassificationLevelOnDuG [-DuGId] <String> [-ClassificationLevelId] <String>
[[Justification] [<String>]] [<CommonParameters>]
```

Table 266: Parameters

Parameter	Description
DuGID	Specify the identifier assigned to the governed resource to be classified (that is, value assigned to UID_QAMDuG parameter). Run the Get-QDataUnderGovernance cmdlet to retrieve a list of governed resources for a managed host or resource path, including their assigned identifiers.
ClassificationLevelId	Specify the identifier assigned to the classification level to be assigned (that is, value assigned to UID_QAMClassificationLevelMan parameter). Run the Get-QClassificationLevelConfiguration cmdlet to retrieve a list of configured classification levels, including their assigned identifiers.
Justification	(Optional) Enter the reason for assigning this classification level

Examples:

Table 267: Examples

Example	Description
Set-QClassificationLevelOnDuG -DuGID 3FAA7F80-F964-4C2A-8F99-045EE43A0A3F -ClassificationLevelId 51442B53-A9BE-4EE0-8A89-B5D5ED3CF387 -Justification "Contains company confidential information"	Manually assign the 'Internal Use Only' classification level (with UID_QAMClassificationLevel value of 51442B53-A9BE-4EE0-8A89-B5D5ED3CF387) to the specified governed resource (with UID_QAMDuG value of 3FAA7F80-F964-4C2A-8F99-045EE43A0A3F).

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales and other inquiries, such as licensing, support, and renewals, visit https://www.oneidentity.com/company/contact-us.aspx.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.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 Ticket
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product

Index