

Google Search Appliance Connectors

Administration Guide

Google Search Appliance Connectors software version 4.1.2

Google Search Appliance software versions 7.4 and 7.6

February 2017



Table of Contents

[Table of Contents](#)

[About this guide](#)

[1 About Connectors 4](#)

[The Lister/Retriever model](#)

[2 What's New in Connectors 4?](#)

[Version 4.1.2](#)

[New SkipDocumentFilter metadata transform](#)

[Support for CIDR notation](#)

[Enhancements to the Connector for Documentum](#)

[Enhancements to the Connector for OpenText](#)

[Version 4.1.1](#)

[Installation includes the Apache Commons Daemon set of applications](#)

[Deprecation of adaptor.domainFormat configuration option](#)

[Enhancements to the Connector for SharePoint](#)

[Enhancements to the Connector for Databases](#)

[Connector for LDAP](#)

[Re-Release of the Connector for Active Directory 4.1.0](#)

[Version 4.1.0](#)

[Connector for OpenText](#)

[Connector for Documentum](#)

[Connector for LDAP](#)

[Access-Controlled serving in secure mode](#)

[New adaptor.domainFormat configuration option](#)

[Enhancements to the Connector for SharePoint](#)

[3 General Information](#)

[Supported connectors](#)

[Supported Google Search Appliance versions](#)

[Supported Java version](#)

[Download the connector software](#)

[Configuration properties file](#)

[Repository content relevancy](#)

[Secure crawling and serving configurations](#)

[Admin Console access](#)

[Required host load changes when running multiple instances on same host](#)

[Host load exception example](#)

[Secure mode](#)

[Supported ACL features](#)

[Mark all documents as public](#)

[Archive feeds](#)

[Reverse proxy setup](#)

[Run a connector as a service on Windows](#)

[Register the connector as a service](#)

[Add optional parameters](#)

[Apache Daemon Logging](#)

[Service Username and Password](#)

[Jvm options](#)

[Run the connector as a service](#)

[Stop running a connector](#)

[4 Enable Connector Security](#)

[Certificate Authorities](#)

[Self-signed certificates](#)

[Create a self-signed certificate for the GSA](#)

[Firefox](#)

[Chrome](#)

[OpenSSL \(command line\)](#)

[Create a self-signed certificate for the connector](#)

[Exchange certificates](#)

[Turn on security with the server.secure property](#)

[Run in secure mode with self-signed certificates](#)

[Enable stricter security](#)

[5 Configure Connector Logs](#)

[Logging properties file](#)

[Change the location of logs](#)

[Change the logging level](#)

[Change the log file size](#)

[Change the number of log files](#)

[6 Monitor Connectors with the Dashboard](#)

[Supported browsers](#)

[Dashboard port number](#)

[Log in to the Connector Dashboard](#)

[View information about a connector](#)

[Version](#)

[Status](#)

[Statistics](#)

[Connector configuration](#)

[Recent log messages](#)

[Start or restart feeds](#)

[Encode sensitive values](#)

[Use the Dashboard](#)

[Use the command-line tool](#)

[Plain text](#)

[Obfuscated](#)

[Encrypted](#)

[Input the sensitive value](#)

[Download rich data about the connector](#)

[7 Troubleshoot Connectors](#)

[Debug a connector by using a web browser](#)

[Troubleshooting scenario](#)

[Troubleshooting quick reference](#)

[8 Common configuration options](#)

About this guide

This Administration Guide is intended for anyone who needs to understand how to manage Google Search Appliance (GSA) Connectors 4.1. It provides overview information about the Connectors, as well as procedures that you can follow to install, configure, or monitor each of the Connectors.

The guide assumes that you are familiar with Windows or Linux operating systems and configuring the Google Search Appliance by using the Admin Console.

For information about installing and configuring particular connectors, see the following guides:

- [Deploying the Connector for Documentum 4.1.2](#)
- [Deploying the Connector for OpenText 4.1.2](#)

These guides, as well as information about using the Admin Console are available from the [Google Search Appliance Help Center](#).

For information about previous versions of connectors, see the [Connector documentation page](#) in the [Google Search Appliance Help Center](#).

1 About Connectors 4

Google Search Appliance connectors enable the Google Search Appliance to acquire content from external repositories and provide that content in search results. A Google Search Appliance with configured connectors can perform fast, unified, secure search across multiple systems and document repositories.

A fundamental strength of the search appliance is discovering enterprise content in web pages and indexing it. The GSA accomplishes this by crawling the web pages over HTTP/HTTPS, following hyperlinks within the pages to interrelated web pages, and adding the content it discovers to the search index. Ultimately, the GSA serves content from its index as search results to end users.

However, many organizations have content that is stored in repositories, such as SharePoint and Windows file shares, rather than on web pages. Because documents in repositories are not usually interrelated through hyperlinks, the search appliance cannot find this content through normal crawling.

Connectors 4 exploit the search appliance's strengths by enabling it to crawl non-web content in repositories over HTTP/HTTPS. Additionally, connectors can feed groups information to the search appliance. Groups information can restrict the visibility of certain content to members of particular groups by using Access Control Lists (ACLs).

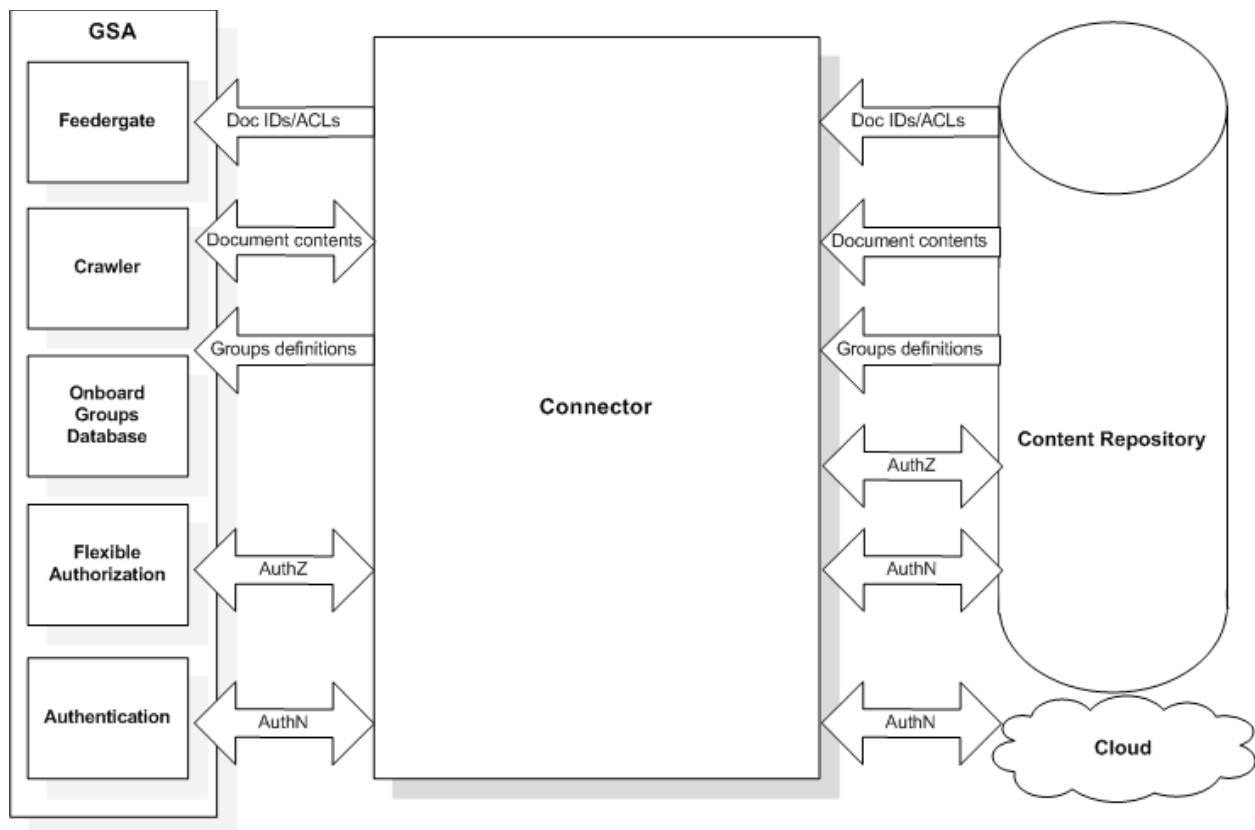
The search appliance adds content acquired through connectors to the search index and uses credentials provided by connectors to protect secure content.

There are several ways to model and communicate your repository's contents to the GSA, and Adaptors are one of them. For other possible solutions, look into [Connectors 3.x](#) and [Content Feeds](#). Connectors 3.x support older GSA versions. Content Feeds should be used when the repository does not provide random document access and instead only provides changes occurring in the repository.

The Lister/Retriever model

Connectors 4 are based on the lister/retriever model. In this model, the lister notifies the search appliance of the names of documents encoded in URLs. The search appliance crawls the URLs and the content is served by the retriever over HTTP/HTTPS.

The following diagram shows how the connectors interact with the search appliance and the repository.



2 What's New in Connectors 4?

Connectors 4 work seamlessly with more search appliance features than previous releases. The following sections describe noteworthy features of Connectors 4 for each release.

Version 4.1.2

Version 4.1.2 includes the following new features:

- [New SkipDocumentFilter metadata transform](#)
- [Support for CIDR notation](#)
- [Enhancements to the Connector for Documentum](#)
- [Enhancements to the Connector for OpenText](#)

New SkipDocumentFilter metadata transform

Version 4.1.2 introduces a new `SkipDocumentFilter` metadata transform, which can be used to exclude documents from indexing based on a given metadata element's values matching or not matching a given regular expression. For detailed information, see the [javadoc](#).

Support for CIDR notation

Version 4.1.2 provides support for CIDR (Classless Inter-Domain Routing) notation in the [server.fullAccessHosts](#) property.

For example, the following value allows access from IP address 10.10.10.10 and the subnet containing the addresses from 192.168.100.0 through 192.168.100.255:

```
server.fullAccessHosts = 10.10.10.10, 192.168.100.0/24
```

Enhancements to the Connector for Documentum

Version 4.1.2 introduces the following enhancement to the Connector for Documentum:

- Set the Last-Modified header so that the last modified date is shown in search results.
- Set a display URL for folders so the precise link will be used in the search results.
- Changed the Content-Type header to use the actual file content type instead of the Documentum Format name (for example, "application/pdf" instead of "pdf"). This affects the value available to metadata transforms, particularly the new [SkipDocumentFilter](#).

- Included the `r_object_id` and `r_object_type` attributes in the metadata by default. The `r_object_type` has multiple values, including the object's own type and all of the supertypes.
- Added new configuration properties:
 - `documentum.indexFolders`--use this property to Indicate whether to index folders and their metadata or not
 - `documentum.documentTypes`--use this property to specify lists of document types to index

For more information, see [Deploying the Connector for Documentum 4.1.2](#).

Enhancements to the Connector for OpenText

Version 4.1.2 introduces the following enhancement to the Connector for OpenText:

- Added support for OpenText Content Web Services with IIS.
- Added support for OpenText Content Server version 16.
- Added support for authenticating against an OpenText Directory Server by using the `opentext.directoryServicesUrl` configuration property.
- Added support for the library configuration property `adaptor.markAllDocsAsPublic`, making it possible to use the connector without admin privileges.
- Added new configuration properties:
 - `opentext.indexFolders`--use this property to Indicate whether to index folders and their metadata or not.
 - `opentext.webServicesServer`--use this property to indicate the type of server (IIS or Tomcat) where Content Web Services is installed.
 - `opentext.windowsDomain`--use this property to specify a domain to be used for all the users and groups with no domain information in the system.

For more information, see [Deploying the Connector for OpenText 4.1.2](#).

Version 4.1.1

Version 4.1.1 introduces the following new features:

- [Installation includes the Apache Commons Daemon set of applications](#)
- [Deprecation of `adaptor.domainFormat` configuration option](#)
- [Enhancements to the Connector for SharePoint](#)
- [Enhancements to the Connector for Databases](#)
- [Connector for LDAP](#)

- [Re-Release of the Connector for Active Directory 4.1.0](#)

Installation includes the Apache Commons Daemon set of applications

The Windows installer for all connectors now includes the [Apache Commons Daemon](#) set of application files (procrun). By using the files in the archive, you can set up the connector to run as a Windows service. For more information, see [Run a connector as a service on Windows](#).

Deprecation of adaptor.domainFormat configuration option

Version 4.1.1 of the Adaptors Library (part of each version 4.1.1 connector) eliminates the necessity of setting the `adaptor.domainFormat` configuration option.

Enhancements to the Connector for SharePoint

Version 4.1.1 introduces enhancement to the Connector for SharePoint, including

- Support for SharePoint 2016
- Support to index multiple site collections. Also, a new configuration flag, `sharepoint.siteCollectionsToInclude`, has been added
- Added 204 support in SharePoint Adaptor for List Items and Files
- Added support for SharePoint URLs with relative redirects.

For more information, see [Deploying the Connector for SharePoint 4.1.1](#).

Enhancements to the Connector for Databases

Version 4.1.1 introduces an enhancement to the Connector for Databases. In this version, it supports a new database mode, `urlAndMetadataLister` mode. In this mode, the GSA retrieves metadata for all URLs. Also, a new configuration flag, `db.includeAllColumnsAsMetadata`, has been added. For more information, see [Deploying the Connector for Databases 4.1.1](#).

Connector for LDAP

Connectors 4.1.1 includes the Connector for Lightweight Directory Access Protocol (LDAP). This connector enables the Google Search Appliance to crawl and index content from an LDAP repository. For more information, see [Deploying the Connector for LDAP 4.1.1](#).

Re-Release of the Connector for Active Directory 4.1.0

Concurrently with the Connector 4.1.1 release, Google is re-releasing the 4.1.0 version of the Connector for Active Directory with a new installer that include the Apache Commons Deamon (procrun). For more information, see “[Run a connector as a service on WIndows](#)” and [Deploying the Connector for Active Directory 4.1.0](#).

Version 4.1.0

Version 4.1.0 introduces the following new features:

- [Connector for OpenText](#)
- [Connector for Documentum](#)
- [Connector for LDAP](#)
- [Access-Controlled serving in secure mode](#)
- New [adaptor.domainFormat configuration option](#)
- [Enhancements to the Connector for SharePoint](#)

Connector for OpenText

Connectors 4.1.0 introduces the Connector for OpenText, which supports OpenText Content Server. For more information, see [Deploying the Connector for OpenText 4.1.0](#).

Connector for Documentum

Connectors 4.1.0 introduces the Connector for Documentum. This new connector enables the Google Search Appliance to crawl and index content from a Documentum repository. For more information, see [Deploying the Connector for Documentum 4.1.0](#).

Connector for LDAP

Connectors 4.1.0 introduces the Connector for Lightweight Directory Access Protocol (LDAP). This new connector enables the Google Search Appliance to crawl and index content from an LDAP repository. For more information, see [Deploying the Connector for LDAP 4.1.0](#).

Access-Controlled serving in secure mode

With connector version 4.1.0, you can configure the database connector and the file system connector to serve access-controlled content to your users by setting up secure mode and using the GSA as a SAML IdP. With access-controlled serving, users can click links and view results in a browser. The connector only serves results that users are allowed to view.

This configuration requires a GSA running software release 7.4, which enables the GSA to act as a SAML Identity Provider (IdP).

For detailed information, see “Access-Controlled serving in secure mode” in either [Deploying the Connector for File Systems 4.1.0](#) or [Deploying the Connector for Databases 4.1.0](#).

New adaptor.domainFormat configuration option

Version 4.1.0 introduces the `adaptor.domainFormat` configuration option. In the following circumstances, the connector needs to combine a name with domain:

- When the connector is using the GSA as an Identity Provider (Idp) and is parsing SAML messages
- When the connector is receiving and parsing GSA authorization requests

An identity for access control check can be in one of the following formats:

- DNS (name@domain)
- NETBIOS (domain\name)
- NETBIOS_FORWARD SLASH (domain/name)
- NONE (name)

To set the format of an identity, use the `adaptor.domainFormat` configuration option, as described in [Common configuration options](#).

Enhancements to the Connector for SharePoint

Version 4.1.0 introduces enhancements to the Connector for SharePoint, including:

- SID (security identifier) based lookup for domain group principals for resolving actual group names from Active Directory
- Leniency in handling unsupported characters in URLs
- Allowing all ADFS (Active Directory Federation Services) claim types to be used in ACLs

For detailed information, see [Deploying the Connector for SharePoint 4.1.0](#).

3 General Information

This section contains general information about Connectors 4, including:

- [Supported connectors](#)
- [Supported Google Search Appliance versions](#)
- [Download the connector software](#)
- [Configuration properties file](#)
- [Repository content relevancy](#)
- [Secure crawling and serving configurations](#)
- [Admin Console access](#)
- [Secure mode](#)
- [Supported ACL features](#)
- [Mark all documents as public](#)
- [Archive feeds on the local drive](#)
- [Reverse proxy setup](#)
- [Run a connector as a service](#)
- [Stop running a connector](#)

Supported connectors

Versions 7.4 and 7.6 of the Google Search Appliance support the following 4.1.2 connectors:

- Connector for Documentum 4.1.2
- Connector for OpenText 4.1.2

Guides for deploying each connector are available from the [Google Search Appliance Help Center](#).

Supported Google Search Appliance versions

Connectors 4.1.2 work with Google Search Appliance version 7.4.0.G.120 or higher.

Supported Java version

The minimum supported versions of Java for Connectors 4.1.2 are JRE 1.7u9 or JRE 8u20. If you want to use the DH (Diffie-Hellman) style of encryption and you are running the GSA with 2048-bit encryption, minimum JRE 8u20 is required.

Download the connector software

Each version 4.1.2 connector must be installed on a host machine. This connector version does not support installing connectors on the Google Search Appliance.

To download the software for a connector, visit

<http://googlelsa.github.io/adaptor/index.html>. Executables are available for all the connectors. Google provides the installation software for each 4.1.2 connector in a single binary file, as listed in the following table.

Repository	Connector	Executable
Documentum	Documentum	documentum-install-4.1.2.exe
OpenText Content Server	OpenText	opentext-install-4.1.2.exe

For information about installing your connector, see the appropriate connector deployment guide, as listed in [About this Guide](#).

Configuration properties file

Configuration is handled in the `adaptor-config.properties` file. For more information about configuration, see the deployment guide that pertains to your connector.

Additionally, there are common configuration variables, which are used by all connectors. If you do not indicate values for these variables, defaults are used. For more information about this topic, see [Common configuration options](#).

Repository content relevancy

The search appliance determines the relevancy in search results of a document that it crawls on the web by using a "pagerank" algorithm, which is based on an analysis of hyperlinks among documents. A search appliance administrator can view the relative pagerank of a document by using the **Index > Diagnostics > Index Diagnostics** page in the Admin Console.

If a document has a content rank, its relative pagerank on the **Index > Diagnostics > Index Diagnostics** page is zero.

The configuration variable `gsa.scoringType` controls whether the search appliance uses a content rank algorithm or pagerank algorithm for repository content. Valid values for this variable are:

- `content` (content rank)
- `web` (page rank)

For each connector, you can specify either the `content` or `web` scoring type. If the connector uses the `web` scoring type, you must add the connector root URL as a **Start URL** on the **Content Sources > Web Crawl > Start and Block URLs** page in the GSA Admin Console. Otherwise, all documents from a connector will have a pagerank of 1.

Secure crawling and serving configurations

Version 4.1.2 connectors support the authentication and authorization configurations for crawling and serving that the GSA administrator configures for the search appliance.

For information about secure crawling and serving configurations, see [Managing Search for Controlled-Access Content](#).

Admin Console access

If the search appliance only allows HTTPS access to the Admin Console, then the connector must be running in secure mode. In secure mode, use HTTPS to access the Connector Dashboard.

To disable HTTPS only access to the Admin Console, select **Enable HTTP (i.e. non SSL) Admin Console and Version Manager access** on the **Administration > System Settings** page in the Admin Console. When HTTP access to the Admin Console is enabled, you can use the Connector Dashboard with or without security enabled for the connector.

Required host load changes when running multiple instances on same host

Host load value (number of concurrent threads indexing content from web server) on the GSA is shared between multiple adaptor instances running on same adaptor host.

To achieve required host load per adaptor instance, use the **Content Sources > Web Crawl > Host Load Schedule** page in the Admin Console:

1. Under **Exception to Web Server Host load**, add host load exception for adaptor host IP Address with host load value as total / sum of host load values desired for each adaptor instance.
2. Under **Exception to Web Server Host load**, add host load exception for individual adaptor instance URLs with desired host load value for that instance.

Host load exception example

Consider 3 adaptor instances running on same host machine (for example, 172.25.52.156) on port 5000, 6000 and 7000 with desired host load value as 4, 4 and 2 respectively.

To achieve desired host load per instance:

1. Add host load exception for IP Address 172.25.52.156 with host load of 10.
2. Add following host load exceptions:
 - http://172.25.52.156:5000/doc/ with host load of 4
 - http://172.25.52.156:6000/doc/ with host load of 4
 - http://172.25.52.156:7000/doc/ with host load of 2

Secure mode

Starting with version 4.1.1 connectors support communication in secure mode over HTTPS. For more information on this topic, see [Enable connector security](#).

Supported ACL features

Access Control Lists (ACLs) control which documents a user can see. The search appliance needs to crawl and index all documents, but still rapidly determine which documents a specific user is allowed to view in a search.

All version 4.1.2 connectors provide ACLs at crawl time, using a separate channel from feeds. Access Control Lists (ACLs) may be inherited from a parent. This reduces the number of ACLs that require re-indexing. Connectors send full-fidelity ACLs, which include inheritance and can contain local groups.

The connectors also support Deny ACLs - ACLs which deny access to specific individuals or groups, local and global namespaces for ACL users and groups and Active Directory groups.

Mark all documents as public

Adding the variable `adaptor.markAllDocsAsPublic=true` to the `adaptor-config.properties` file makes the connector identify each document as visible by everyone. The default value for `adaptor.markAllDocsAsPublic` is "false."

Archive feeds

Adding the variable `feed.archiveDirectory` with a valid path to the `adaptor-config.properties` file enables you to save feeds to the specified directory on the local drive as they are sent to the GSA. All feeds successfully and unsuccessfully sent to the GSA are archived. Failed feeds are tagged with FAILED in the archive feed file name. The feeds contain listed document-ids, named resources, and group definitions.

Reverse proxy setup

You can add a reverse proxy to your configuration as an intermediary for crawl requests from the search appliance to the connectors. For example, you might configure a proxy server (`PROXYHOST`) between the search appliance and multiple connectors in a round-robin setup.

To set up a reverse proxy:

1. Configure `server.hostname` as the proxy server instead of the server running the connector. In the previous example, you would configure `server.hostname=PROXYHOST`
2. Optionally configure `server.reverseProxyPort` (defaults to `server.port`). This option controls the port used in retriever URLs.
3. Optionally configure `server.reverseProxyProtocol` to either `http` or `https`, depending on proxy traffic (defaults to `https` in secure mode and `http` otherwise).

Run a connector as a service on Windows

When you run a connector as a service, you do not need to run it manually. The connector runs when the host server starts up, and shuts down when the host is shut down. Before running the connector as a service, register it, as described in the following section.

Register the connector as a service

You register the connector as a service by running the `prunsv` command -- made by Apache -- as shown in the following procedure. The Windows installer for the connector creates a directory called `<installation home>/tools/procrun`, unzips the Apache Commons Daemon zip archive, and installs the following executable files:

- `<installation home>/tools/procrun/prunsv.exe--32-bit version`
- `<installation home>/tools/procrun/amd64/prunsv.exe--x86-64-bit version`
- `<installation home>/tools/procrun/ia64/prunsv.exe--Intel Itanium 64-bit version`

Take note that you can increase the power of the `prunsv` command by adding optional parameters. For example, you can specify logging for the Apache daemon by using the `LogPath` parameter. For detailed information, see "[Add optional parameters.](#)"

To register a connector as a service:

1. Copy the correct version of `prunsv.exe` for your machine from the `<installation home>/tools/procrun/` subdirectory and put it in the same directory as the connector you would like to run as a service.
2. In the same directory where the connector `.jar` files are installed, run the following command:

```
prunsv install <CONNECTOR-NAME> ^
--StartPath="<STARTPATH>" ^
--Classpath=<CONNECTOR-JAR> ^
--StartMode=jvm ^
--StartClass=com.google.enterprise.adaptor.Daemon ^
--StartMethod=serviceStart ^
--StartParams=<FULL-CONNECTOR-CLASSNAME> ^
--StopMode=jvm ^
--StopClass=com.google.enterprise.adaptor.Daemon ^
--StopMethod=serviceStop ^
--StdOutput=<OUTPUT-LOG> ^
--StdError=<ERROR-LOG> ^
--Jvm=<JVM-DLL> ^
--Startup=auto
```

Where:

<CONNECTOR-NAME> is the name of the connector in the list of running services:

Documentum: adaptor-documentum

OpenText: adaptor-opentext

<STARTPATH> is the absolute path of the StartPath, for example

"C:\Users\administrator.GSA\Desktop\Connector"

<CLASSPATH> is the the name of the connector .jar, for example, adaptor-sharepoint-4.1.2-withlib.jar

For the Documentum connector, you must include the path to dctm.jar and the Documentum config directory in the classpath, for example,

```
--Classpath="<path_to_dctm.jar>;<path to Documentum config
directory>;adaptor-documentum-4.1.2-withlib.jar" ^
```

<FULL-CONNECTOR-CLASSNAME> is one of the following values:

com.google.enterprise.adaptor.documentum.DocumentumAdaptor

com.google.enterprise.adaptor.opentext.OpenTextAdaptor

<OUTPUT-LOG> is the full path to the output log, for example,

C:\sp\logs\stdout.log

<ERROR-LOG> is the full path to the error log, for example,

C:\sp\logs\stderr.log

<JVM-DLL> is the path to where Java Virtual Machine dynamic link library is installed, for example, C:\Java\jdk1.7u6\jre\bin\server\jvm.dll

An alternative to specifying the JVM on the command line with the `Jvm` parameter is to configure the default JVM with the Java Control panel (`javacpl.exe`). Be sure to update the service registration each time you update the JVM.

Add optional parameters

You can add important optional parameters to the `prunsrv` command to specify:

- Apache daemon logging
- Service username and password
- JVM options

Apache Daemon Logging

You can specify logging for the Apache Daemon by using the `LogPath` parameter. The default value is:

```
%SystemRoot%\System32\LogFiles\Apache
```

where `SystemRoot` is a root path, for example `C:\Windows`

Service Username and Password

Use the `ServiceUser` parameter to specify the name of the account under which the service should run, as shown in the following example:

```
--ServiceUser DOMAINname\username ^
```

Use the `ServicePassword` to specify the password for the account designated by the `ServiceUser` parameter, as shown in the following example:

```
--ServicePassword password ^
```

Jvm options

Use the `JvmOptions` parameter to specify a `JvList` of options in the form of `-D` or `-X` that will be passed to the JVM, as shown in the following example:

```
++JvmOptions=-Djava.util.logging.config.file=logging.properties
```

Run the connector as a service

To run a connector as a service, run the following command in the same directory where the connector .jar files are installed:

```
prunsrv start <CONNECTOR-NAME>
```

Where `<CONNECTOR-NAME>` is the internal name of the connector:

- Documentum: `adaptor-documentum`
- OpenText: `adaptor-opentext`

Stop running a connector

To stop running a connector in Windows or Linux, close the connector command prompt on the host.

You can stop running a connector as a service on Windows from either the service list or the command line.

To stop running a connector as a service on Windows from the service list:

1. On the connector host, choose **Start > Run > services.msc**.
2. Select the connector service.
3. Click **Stop**.

To stop running a connector as a service on Windows from the command line, enter the following command on the host:

```
prunsrv stop <CONNECTOR_NAME>
```

4 Enable Connector Security

In secure mode, the connectors communicate with the Google Search Appliance over HTTPS. You can enable security for any connector by configuring certificates and turning on security.

Secure mode supports using either of the following types of certificates:

- [Certificate Authorities \(CA's\)](#)
- [Self-signed certificates](#)

In either case, you can also choose options to [enable stricter security](#).

Certificate Authorities

The GSA and the connector executable both have default Certificate Authorities; public keys are already in the GSA and connector trust stores. For the connector, you can find the default keystore CAs under `jre\lib\security\`.

If you are using the default CA's only, complete the tasks described in the following sections:

- [Exchange certificates](#)
- [Turn on security](#)

By default, the search appliance alias is "gsa" and the connector alias is "adaptor." Optionally, you can configure either alias.

Self-signed certificates

If you need to create self-signed certificates before turning on security, complete the tasks described in the following sections:

- [Create a self-signed certificate for the GSA](#)
- [Create a self-signed certificate for the connector](#)
- [Exchange certificates](#)
- [Turn on security with the `server.secure` property](#)

Create a self-signed certificate for the GSA

For information about creating a self-signed certificate for the search appliance, see the GSA Admin Console help page for [Administration > SSL Settings](#).

To get the GSA's freshly-created certificate to add it as a trusted host for the connector, follow the procedure for your preferred browser or the command line.

Firefox

1. Navigate to the GSA's secure search: `https://gsahostname/`.
A warning page appears with the following message: "This Connection is Untrusted."
This message appears because the certificate is self-signed and not signed by a trusted Certificate Authority.
2. Click, "I Understand the Risks" and "Add Exception."
3. Wait until the "View..." button is clickable, then click it.
4. Change to the "Details" tab and click "Export...".
5. Save the certificate in your connector's directory with the name "gsa.crt".
6. Click **Close** and **Cancel** to close the windows.

Chrome

1. Navigate to the GSA's secure search: `https://gsahostname/`.
A warning page appears with the following message: "The site's security certificate is not trusted!" In the location bar, there should be a padlock with a red 'x' on it.
2. Click the padlock and then click "Certificate Information."
3. Change to the "Details" tab and click "Export...".
4. Save the certificate in your adaptor's directory with the name "gsa.crt".
5. Click **Close** and **Cancel** to close the windows.

OpenSSL (command line)

1. Execute the following command:

```
openssl s_client -connect gsahostname:443 < /dev/null
```
2. Copy the section that begins with `-----BEGIN CERTIFICATE-----` and ends with `-----END CERTIFICATE-----` (including the `BEGIN` and `END CERTIFICATE` portions) into a new file.
3. Save the file in your connector's directory with the name "gsa.crt".

Create a self-signed certificate for the connector

Generate a self-signed certificate for the connector and export the newly created certificate.

1. Within the connector's directory, run the following command:

```
keytool -genkeypair -keystore keys.jks -storepass changeit  
-keypass changeit -alias adaptor -keyalg RSA -validity 365
```
2. For "What is your first and last name?", enter the hostname of the connector's computer. You are free to answer the other questions however you wish (including not answering them).
3. Answer "yes" to "Is CN=yourcomputershostname, OU=... correct?"
4. Still in connector's directory, run the following command:

```
keytool -exportcert -alias adaptor -keystore keys.jks -storepass  
changeit -keypass changeit -rfc -file adaptor.crt
```
5. Copy cacerts from Java to the connector's directory:

For Windows, run the following command:

```
copy PATH\TO\JRE\lib\security\cacerts cacerts.jks
```

For Linux ,run the following command:

```
cp PATH/TO/JRE/lib/security/cacerts cacerts.jks
```

6. To allow the connector to trust itself, run the following command:

```
keytool -importcert -keystore cacerts.jks -storepass changeit  
-file adaptor.crt -alias adaptor
```
7. When prompted **Trust this certificate?**, answer yes.

Exchange certificates

To allow the connector to trust the search appliance:

1. On the connector host, run the following command:

```
keytool -importcert -keystore cacerts.jks -storepass changeit  
-file gsa.crt -alias gsa
```
2. When prompted **Trust this certificate?**, answer yes.

To allow the search appliance to trust the connector:

1. In GSA Admin Console, click **Administration > Certificate Authorities**.
2. Under **Add more Certificate Authorities**, click **Browse**.
3. Navigate to the connector's directory and select adaptor.crt.
4. Click **Save**.

Turn on security with the `server.secure` property

You can turn on security for the connector by using `server.secure` property, which enables HTTPS and certificate checking. Add the following line to your `adaptor-config.properties` file:

```
server.secure=true
```

When `server.secure=true`, the connector uses the GSA's authentication configuration and HTTPS for all communication. Also, when the value of `server.secure` is `true`, the following conditions apply:

- You need to add the key to the connector keystore with an alias defined in the connector config file, `server.keyAlias=adaptor`
- The connector runs on the configured port enforcing SSL.
- The [Connector Dashboard](#) runs on the configured port enforcing SSL.
- Feeds from the connector are forced to the search appliance secure FeederGate port (19902), even if the search appliance accepts feeds over HTTP.
- The connector validates the search appliance's certificate during the SSL handshake.

Run in secure mode with self-signed certificates

If you are using one or more self-signed certificates in your configuration, you must run the connector with SSL settings, as shown in the following example command:

(Windows):

```
java ^
-Djava.util.logging.config.file=src/logging.properties ^
-Djavax.net.ssl.keyStore=keys.jks ^
-Djavax.net.ssl.keyStoreType=jks ^
-Djavax.net.ssl.keyStorePassword=changeit ^
-Djavax.net.ssl.trustStore=cacerts.jks ^
-Djavax.net.ssl.trustStoreType=jks ^
-Djavax.net.ssl.trustStorePassword=changeit ^
-classpath adaptor-name-4.1.2-withlib.jar ^
com.google.enterprise.adaptor.name.NameAdaptor
```

(Linux / Unix systems):

```
java \
-Djava.util.logging.config.file=src/logging.properties \
-Djavax.net.ssl.keyStore=keys.jks \
-Djavax.net.ssl.keyStoreType=jks \
-Djavax.net.ssl.keyStorePassword=changeit \
-Djavax.net.ssl.trustStore=cacerts.jks \
-Djavax.net.ssl.trustStoreType=jks \
-Djavax.net.ssl.trustStorePassword=changeit \
-classpath adaptor-name-4.1.1-withlib.jar \
com.google.enterprise.adaptor.name.NameAdaptor
```

Enable stricter security

Optionally, you can improve security by choosing stricter security features on the **Administration > SSL Settings** page in the Admin Console, as described in the following table. However, using any of these options require the connector to be configured for security and have `server.secure=true` in its configuration.

Option	Setting	Description
Enable HTTP (non-SSL) access for FeederGate	Uncheck	When this option is unchecked, only HTTPS communications will be accepted by FeederGate. Connectors send document ids to FeederGate.
Enable Client Certificate Authentication for FeederGate	Check	When this option is checked, the FeederGate SSL port (19902) only accepts connections from IP addresses in the trusted IP addresses list and clients who present a valid x509 certificate when connecting. Valid means that the certificate is signed by a certificate in the CA keystore on the search appliance (or a certificate in the certificate chain).
Enable Server Certificate Authentication	Check	When this option is checked, it is a requirement for the crawler to authenticate certificates presented by servers that contain secure content.

You must include `server.secure=true` in the connector configuration before enabling these stricter features.

To enable stricter security, perform the following steps by using the GSA Admin Console:

1. Click **Administration > SSL Settings**.
2. Make any of the following changes on this page:
 - a. Uncheck **Enable HTTP (non-SSL) access for FeederGate**.
 - b. Check **Enable Client Certificate Authentication for FeederGate**.
 - c. Check **Enable Server Certificate Authentication**.
3. Click **Save**.

5 Configure Connector Logs

The connectors log processing messages, including exceptions and warnings. Log messages appear in the [Connector Dashboard](#) and you can download the logs, as described in [Download rich data about the connector](#).

Messages contain information about thread processing, including:

- Date stamp--Date and time the message was logged
- Name-of-thread--The thread that generated the message
- Last-30-characters-of-method--Code source for connector request
- Logging level--Filter log messages by level of severity
- Log-message--Text message for log entry

The following example shows a log message:

```
06-12 18:20:08.839 background URLConnection.getInputStream() FINE:
sun.net.www.MessageHeader@b20ccbf12 pairs:...
```

Logging properties file

Log configuration is controlled by the `logging.properties` file. Each connector installation procedure in this documentation contains a step for editing `logging.properties`. By editing values in this file, you can configure the following settings:

- [Location of logs](#)
- [Logging level](#)
- [Log file size](#)
- [Number of log files](#)

The following example shows a `logging.properties` file with default values.

```
handlers = java.util.logging.FileHandler
.level = WARNING
com.google.enterprise.adaptor.level = INFO
java.util.logging.FileHandler.formatter=com.google.enterprise.adaptor.
CustomFormatter
java.util.logging.FileHandler.pattern=logs/adaptor.%g.log
java.util.logging.FileHandler.limit=104857600
java.util.logging.FileHandler.count=20
```

Change the location of logs

By default, the logs are saved in `logs/adaptor.*.log`, in the same directory where the connector is running.

To change the location of log files, edit the `java.util.logging.FileHandler.pattern` value in the `logging.properties` file:

```
java.util.logging.FileHandler.pattern=logs/adaptor.%g.log
```

Change the logging level

You can filter messages written to log files by the following Java log levels:

- FATAL
- WARNING
- INFO
- FINE
- FINER
- FINEST

By default, the log level is INFO. The number of messages generated increases with each level, where FATAL logs the smallest number of messages and FINEST logs the largest.

To change the level of log files, edit the `com.google.enterprise.adaptor.level` value in the `logging.properties` file:

```
com.google.enterprise.adaptor.level=FINE
```

Change the log file size

By default, the size of connector log files is 10485760 bytes. Restarting the connector will create a new log file, regardless of how large the previous one had been.

You can change the size to suit your needs. The limit must be specified as a 32-bit integer, and thus has an upper limit of 2,147,483,647 (2 gigabytes, about 205 times as large as the default size).

To change the size of log files, edit the `java.util.logging.FileHandler.limit` value in the `logging.properties` file:

```
java.util.logging.FileHandler.limit=10485760
```

Change the number of log files

The connector writes to a log file until the size limit is reached, then starts writing to a new log file. By default, the connector writes to 20 log files, but you can change the number to suit your needs. There is no upper limit to the number of log files. After it finishes writing to the last log file, it starts writing over the first file.

To change the number of log files, edit the `java.util.logging.FileHandler.count` value in the `logging.properties` file:

```
java.util.logging.FileHandler.count=20
```

6 Monitor Connectors with the Dashboard

The Dashboard is a web-based interface that provides information about the connector's operation, with easy access to logs and error history.

Use the Connector Dashboard to perform the following tasks:

- [View information about the connector](#)
- [Start or restart feeds](#)
- [Encode sensitive values](#)
- [Download rich data about the connector](#)

You must start the connector to use the Dashboard.

Supported browsers

The Connector Dashboard runs in the following browsers:

- Google Chrome 22
- Internet Explorer 8 and 9
- Firefox 15 and 16
- Safari 5 and 6

Dashboard port number

Windows installers default to a different value for each connector:

- 6279 for the Connector for Documentum
- 6379 for the Connector for OpenText

The port number is determined by the value of the variable `server.dashboardPort` in the `adaptor-config.properties` file for the connector. You can change the Connector Dashboard port number by changing the default value in this file. Every instance of a connector running on a host must have a unique value for `server.dashboardPort`.

Log in to the Connector Dashboard

To display the Connector Dashboard, open a browser and navigate to the following HTTP or HTTPS address:

`http://<CONNECTOR_HOST>:<nnnn>/dashboard`

or

`https://<CONNECTOR_HOST>:<nnnn>/dashboard`

where:

- HTTP or HTTPS--If you run the connector in [secure mode](#), use HTTPS to log in to the Dashboard.
- <CONNECTOR_HOST> is the hostname or IP address of the host that is running the connector.
- <nnnn> is the dashboard port number, as specified in the `adaptor-config.properties` file for the connector.

To log in to the Connector Dashboard, use your search appliance user or administrator login credentials. You cannot log in to the Connector Dashboard with search appliance manager login credentials.

View information about a connector

You can use the Dashboard to monitor the connector by viewing up-to-date information, including:

- [Version](#)
- [Status](#)
- [Statistics](#)
- [Connector \(Adaptor\) configuration](#)
- [Recent log messages](#)

Version

In the Version section, the Dashboard displays information about the currently installed Java version, Connector ("Adaptor") library version, Connector type, and Connector version.

Status

In the Status section, the Connector Dashboard displays the current status of the Java version (supported or not), feed pushing, the error rate of document retrieval from the repository (derived from logs), and search appliance crawling.

For each item, a signal indicates the status by color:

- Green for OK. The item is functioning.
- Yellow for alert. The item is not currently functioning, but no action is required. For example, the Dashboard displays yellow when the GSA is not currently crawling.
- Red for warning. The item is not functioning and requires attention.

Statistics

In the Statistics section, the Connector Dashboard displays the following information:

- A datestamp for when the connector program was started.
- Datestamps for the last successful push (full or incremental) start or end. The push can either be started automatically or manually.
- Status of the current push, if any.
- Total number of DocIDs pushed from the repository to the connector since the program started.
- Total number of requests for documents and unique documents from the GSA and the connector.
- Time resolution

The Statistics section also displays graphs showing throughput and response time for the last minute, last hour, and last day.

Connector configuration

In the configuration section, the Connector Dashboard displays the values for all the configuration variables in the `adaptor-config.properties` file. It also displays default values for any property not overridden in the `adaptor-config.properties` file.

Recent log messages

In the Recent Log Messages section, the Dashboard displays connector log messages. For more information on this topic, see [Download rich data about the connector](#).

Start or restart feeds

The Connector Dashboard enables you to start or restart a full feed or an incremental feed as often as needed or when errors are detected. To start or restart a feed, click either **Run Incremental Push** or **Run Full Push**.

Encode sensitive values

You can encode passwords and other sensitive configuration values and copy them to the `adaptor-config.properties` file. Values can be specified in the configuration as `prefix:data`, where the prefix specifies how the value is stored.

You can encode the listed sensitive values for the following connectors:

- Connector for Documentum--documentum.password
- Connector for OpenText--opentext.password

Take note that the Connector for File Systems does not store sensitive values. However, it supports two secure modes of operation:

- Running the connector as a privileged user with access to the file systems.
- Mounting the remote file systems using NET USE /USER: before starting the connector.

The sensitive value can be stored as:

- **Plain text** allowing the password or other information to be read by anybody who can read the configuration. Denoted by "pl" prefix.
- **Obfuscated** where the information is in a highly unreadable format, but it is possible for anyone to retrieve the original text. Denoted by "obf" prefix.
- **Encrypted** which uses your HTTPS encryption key to encrypt the value. Denoted by "pkc" prefix.

To encode a sensitive value, you can use either:

- [The Dashboard](#) or
- [The command-line tool](#)

Use the Dashboard

To use the Dashboard to encode a sensitive value:

1. Under **Storing Sensitive Values**, enter the sensitive value in the field.
2. Click a storage option.
3. Click **Encode Sensitive Value**.
The encoded value appears.
4. Copy and paste the sensitive value into the `adaptor-config.properties` file.

Use the command-line tool

To use the command-line tool to encode a sensitive value , run one of the following commands, depending on how you want the value stored:

- Plain text
- Obfuscated
- Encrypted

Plain text

The following command example shows how to encode a value and store it as plain text:

```
java \  
-classpath adaptor-4.1.2-withlib.jar \  
com.google.enterprise.adaptor.SensitiveValueCodec \  
-DsecurityLevel=PLAIN_TEXT
```

Obfuscated

The following command example shows how to encode a value and store it as obfuscated:

```
java \  
-classpath adaptor-4.1.2-withlib.jar \  
com.google.enterprise.adaptor.SensitiveValueCodec \  
-DsecurityLevel=OBFUSCATED
```

Encrypted

The following command example shows how to encode a value and store it as encrypted:

```
java \  
-Djavax.net.ssl.keyStore=keys.jks \  
-Djavax.net.ssl.keyStoreType=jks \  
-Djavax.net.ssl.keyStorePassword=changeit \  
-classpath adaptor-4.1.2-withlib.jar \  
com.google.enterprise.adaptor.SensitiveValueCodec \  
-DsecurityLevel=ENCRYPTED \  
-Dserver.keyAlias=adaptor \  
-Dserver.secure=true
```

Input the sensitive value

After running this command, you are prompted to input the sensitive value (the sensitive value you enter will not reflect on command line interface for the sake of security). When finished, press Enter and the encoded value is displayed.

Download rich data about the connector

The Diagnostics zip archive contains rich data about the connector, including:

- Current configuration settings (in the `config.txt` file)
- Connector version, status, and statistics (in the `state.txt` file)
- Thread details (in the `threaddump.txt` file)
- Logs folder

This data that can help diagnose connector issues. To download the archive, click **Diagnostics zip file** on the Dashboard.

7 Troubleshoot Connectors

Version 4.1.1 connectors provide several options for troubleshooting issues, including:

- [Connector Dashboard](#) for checking the status of feeds and document retrieval
- [Logs on connector machine](#) for checking messages about thread processing
- [Search appliance index diagnostics](#) for checking crawl status
- [Search appliance real-time diagnostics](#) for checking HTTP headers for a specific URL at any time without having to wait for the crawler to ingest it
- [Web browser](#) with the connector host

Additionally, you can troubleshoot issues by examining URL-and-metadata feed files.

Debug a connector by using a web browser

A connector, by default, will deny all document accesses, except from the search appliance. To allow debugging and testing a connector by using a browser without a search appliance, you can add a hostname to the `server.fullAccessHosts` configuration option to allow that computer full access to all connector content.

In addition, this setting allows that computer to see metadata and other GSA-specific information as HTTP headers. This capability can be very useful when combined with Firebug or the Web Inspector in your browser to observe a connector's behavior.

Troubleshooting scenario

In this scenario, users cannot find a specific document in search results, even though it is assumed to be in the search appliance index. To troubleshoot this issue, the administrator can track the document through the system by following the path a document takes to get into the search appliance index.

The administrator might perform one or more of the following steps:

1. Make sure that the search appliance is set to follow and crawl the Connector's URLs by checking the **Content Sources > Web Crawl > Start and Block URLs** page in the Admin Console.
2. Make sure GSA crawling is not paused by using the **Content Sources > Diagnostics > Crawl Status** page.
3. Check the Connector status and recent log messages by using the Dashboard.

4. Ensure that the Connector fed the document URL to the search appliance by examining the feed file.
Starting with Adaptors v4.1.1, the feed file contains comments that indicate the Adaptor Library version, the version of the Adaptor, and the version of Java being used. If you do not see that information near the top of the feed file (typically lines 4-6 of the feed), then the adaptor that sent the feed is out of date.
5. Ensure that the connector logged the document by checking connector log files. The Lister logs a file when it feeds it to the search appliance. The Retriever logs the file when the crawler requests the document.
6. Check the HTTP header for the document by using the **Content Sources > Diagnostics > Real-time Diagnostics** page in the Admin Console.
7. Find out if the connector has information about the document by using a web browser to access the file information on the connector host.
8. Ensure that the search appliance got the document by using the **Index> Diagnostics > Index Diagnostics** page in the Admin Console.

If the document isn't located, the administrator can request a recrawl of the missing document by restarting the crawl from the Connector Dashboard, or recrawling the URL by using the **Content Sources > Web Crawl > Freshness Tuning** page in the Admin Console.

Troubleshooting quick reference

Error message/Issue	Resolution	Type of Connector
Logs: Unauthorized request. Status code:200	Add host IP to the GSA's feeds' list of trusted IP addresses by using the Content Sources > Feeds page in the Admin Console.	SharePoint, SharePoint User Profiles, File Systems
Index diagnostics: Error: 400 Bad request. The GSA is trying to crawl relative links that are invalid in indexed documents.	Ensure that the connector URL ends with <code>"/doc/"</code> by using the Content Sources > Web Crawl > Start and Block URLs, Follow Patterns . For example: <code>http://connector.example.com:6278/doc/</code>	All

Index diagnostics: Error: Permanent DNS failure.	Add a DNS override by using the Administration > DNS Override page in the Admin Console.	SharePoint
Index diagnostics: Retrying URL: Connection reset by peer during fetch.	DNS override is wrong. Correct it by using the Administration > DNS Override page in the Admin Console.	SharePoint
Errors in the logs for some documents:The server sent HTTP status code 503: Service unavailable	Host load is too high, try to reduce host load	SharePoint
Feeds are not coming through	<ul style="list-style-type: none"> • Make sure GSA can accept feeds from the connector host machine. • Check connector logs for errors, such as failure to look-up GSA, or failure to communicate with the repository. 	SharePoint, SharePoint User Profiles, File Systems, Active Directory
Documents are not getting indexed	<ul style="list-style-type: none"> • Make sure GSA is set to follow and crawl the Connector's URLs by checking the Content Sources > Web Crawl > Start and Block URLs page in the Admin Console. • Make sure GSA crawling is not paused by using the Content Sources > Diagnostics > Crawl Status page. • Check for error 	SharePoint, SharePoint User Profiles, File Systems

	<p>messages on the Index> Diagnostics > Index Diagnostics page.</p> <ul style="list-style-type: none"> • Take a look at connector's log messages. • Check the Content Sources > Diagnostics > Real-time Diagnostics page for the particular URL that you expect to be indexed. 	
Crawling is slow	<p>Use the Dashboard to find:</p> <ul style="list-style-type: none"> • What is the mean duration of a request (Response Time)? A couple hundred milliseconds would be good. • What is the max duration of a request? A file taking over a couple of minutes would be bad. 	SharePoint, File Systems
Document retrieval times out	<p>The connector gives a document retrieval request 10 minutes to send document headers, and 10 minutes from that time to send the content. If you want to give your repository more time, you can adjust <code>adaptor.docContentTimeoutSecs</code> &</p>	SharePoint, File Systems

	<code>adaptor.docHeaderTimeoutSecs .</code>	
The Google Search Appliance Index Diagnostics shows many documents with a Crawl Status of "Document not found (404)."	Files and folders that are marked as hidden are not fed to the GSA. However, they may be listed on the Index> Diagnostics > Index Diagnostics page with a crawl status of "Document not found (404)."	SharePoint, File Systems
SharePoint is returning 401 (unauthorized)	Ensure that the full read permissions are given on the SharePoint Web Application policy.	SharePoint
Renamed user names are not reflected in ACLs.	Run User Profile Synchronization job for incremental updates.	SharePoint

8 Common configuration options

The following table lists common configuration options, which are used by all connectors. If the administrator doesn't set these options, defaults are used. The only required option is `gsa.hostname`. All others are optional.

Name	Meaning	Default
<code>adaptor.docContentTimeoutSecs</code>	Number of seconds a connector has to complete sending content before it is interrupted. Timing starts when sending content starts.	600
<code>adaptor.docHeaderTimeoutSecs</code>	Number of seconds connector has to start sending content before it is interrupted.	600
<code>adaptor.domainFormat</code> (Note: in version 4.1.1, this option is deprecated)	Indicates that ACL names are in one of the following formats: <ul style="list-style-type: none">• DNS (name@domain.com)• NETBIOS (domain\name)• NETBIOS_FORWARDSLASH (domain/name)• NONE (name)	DNS
<code>adaptor.disableFullAndIncrementalListing</code>	When the value is "true," the adaptor functions only as a retriever, not as a listener. Also, when "true," the values of the <code>adaptor.pushDocIdsOnStartup</code> , <code>adaptor.fullListingSchedule</code> , and <code>adaptor.incrementalPollPeriodSecs</code> properties are ignored and the full and incremental feed buttons on the dashboard are disabled.	false

<code>adaptor.fullListingSchedule</code>	When to invoke <code>adaptor.getDocIds</code> , in cron format (minute, hour, day of month, month, day of week).	0 3 * * *
<code>adaptor.heartbeatTimeoutSecs</code>	Number of seconds a request for a "heartbeat" URL is allowed to take, before it times out.	30
<code>adaptor.incrementalPollPeriodSecs</code>	Number of seconds between invocations of <code>PollingIncrementalLister.getModifiedDocIds</code> .	900
<code>adaptor.markAllDocsAsPublic</code>	When the value is "true," all documents are marked as "public."	false
<code>adaptor.namespace</code>	If a username or group contains a Windows domain, the value of <code>adaptor.namespace</code> will be used for the namespace when sending the user/group to the GSA. Otherwise, the namespace will be constructed from <code>adaptor.namespace</code> and the hostname where Content Server is running.	
<code>adaptor.pushDocIdsOnStartup</code>	Whether to invoke <code>adaptor.getDocIds</code> on process start in addition to <code>adaptor.fullListingSchedule</code> .	true
<code>docId.isUrl</code>	If your connector document ids are already URLs, prevent them from being inserted into connector generated URLs.	false
<code>feed.archiveDirectory</code>	Save feeds of listed document-ids, named resources, and group	

	definitions to the specified directory on the local drive as they are sent to the GSA. All feeds successfully and unsuccessfully sent to the GSA are archived. Failed feeds are tagged with FAILED in the archive feed file name.	
<code>feed.crawlImmediatelyBitEnabled</code>	Send bit telling GSA to crawl immediately.	false
<code>feed.maxUrls</code>	Set the maximum number of URLs included per feed file.	5000
<code>feed.name</code>	Source name used in feeds. Generated if not provided.	
<code>feed.noRecrawlBitEnabled</code>	Send bit telling the GSA to crawl your documents only once.	false
<code>gsa.acceptsDocControlsHeader</code>	Use X-Gsa-Doc-Controls HTTP header with namespaced ACLs. Otherwise ACLs are sent without namespace and as metadata. If not set, then an attempt to compute from gsa.version is made.	Determined by examining the GSA version
<code>gsa.characterEncoding</code>	Character set used in feed files.	UTF-8
<code>gsa.hostname</code>	Machine to send feed files to. Connector execution halts with error if not provided.	
<code>gsa.samlEntityId</code>	The SAML Entity ID that identifies the GSA.	http://google.com/enterprise/gsa/T3-LF7R76PVGW2T4
<code>gsa.scoringType</code>	Type of relevance algorithm GSA utilizes to rank documents.	content

	Either content or web. Is sent when <code>adaptor.acceptsDocControlHeader</code> is true.	
<code>gsa.version</code>	Version number used to configure expected GSA features.	Defaults to acquiring from GSA. Uses 7.0.14-114 if acquiring fails.
<code>journal.reducedMem</code>	Avoid tracking per URL information in RAM; suggested with over five hundred thousand documents.	true
<code>server.dashboardPort</code>	Port on a connector's machine for accessing a connector's dashboard. Every instance of a connector running on a machine must have a unique value for <code>server.dashboardPort</code> .	5679
<code>server.docIdPath</code>	Part of URL preceding encoded document ids.	/doc/
<code>server.fullAccessHosts</code>	Hosts allowed access without authentication (certificates still needed when in secure mode).	empty, but implicitly contains <code>gsa.hostname</code>
<code>server.heartbeatPath</code>	Part of URL preceding (content-availability-only) encoded document ids.	/heartbeat/
<code>server.hostname</code>	Hostname of a connector machine for URL generation. The GSA will use this hostname to crawl the connector.	lowercase of automatically detected hostname
<code>server.keyAlias</code>	Keystore alias where encryption (public and private) keys are	adaptor

	stored.	
<code>server.maxWorkerThreads</code>	Number of maximum simultaneous retrievals allowed.	16
<code>server.port</code>	Retriever port. Every instance of a connector running on a machine must have a unique value for <code>server.port</code> .	5678
<code>server.queueCapacity</code>	Maximum retriever queue size.	160
<code>server.reverseProxyPort</code>	Port used in retriever URLs (in case requests are routed through a reverse proxy).	<code>server.port</code>
<code>server.reverseProxyProtocol</code>	Can be either http or https, depending on proxy traffic.	https in secure mode or http otherwise
<code>server.samlEntityId</code>	The SAML Entity ID that identities connector.	<code>http://google.com/enterprise/gsa/adaptor</code>
<code>server.secure</code>	Enables https and certificate checking.	false
<code>server.useCompression</code>	Compress retrieval responses.	true
<code>transform.acl.X</code>	Where X is an integer, match and modify principals as described.	no modifications
<code>transform.pipeline</code>	Sequence of transformation steps.	empty string (no pipeline)