# Oracle® Endeca Server

Security Guide

Version 7.6.1 • June 2014 • Revision A

ORACLE

# Copyright and disclaimer

# Table of Contents

# Preface

Oracle® Endeca Server is a hybrid search-analytical engine that organizes complex and varied data from disparate sources. At the core of Endeca Information Discovery, the unique NoSQL-like data model and in-memory architecture of the Endeca Server create an extremely agile framework for handling complex data combinations, eliminating the need for complex up-front modeling and offering extreme performance at scale. Endeca Server also supports 35 distinct languages.

# About this guide

This guide describes the Oracle Endeca Server security features and the major tasks involved in using them to develop a secure Endeca implementation.

# Who should use this guide

This guide is for developers who are responsible for implementing security features in Endeca applications.

This guide assumes that the Oracle Endeca Server software is already installed. It also assumes that you are familiar with software security topics, such as digital certificates, certificate authorities, and mutual authentication.

# Conventions used in this guide

The following conventions are used in this document.

## Typographic conventions

This table describes the typographic conventions used when formatting text in this document.

| Typeface | Meaning |
|---|---|
| **User Interface Elements** | This formatting is used for graphical user interface elements such as pages, dialog boxes, buttons, and fields. |
| `Code Sample` | This formatting is used for sample code phrases within a paragraph. |
| *Variable* | This formatting is used for variable values.<br>For variables within a code sample, the formatting is *`Variable`*. |
| `File Path` | This formatting is used for file names and paths. |

## Symbol conventions

This table describes the symbol conventions used in this document.

| Symbol | Description | Example | Meaning |
|---|---|---|---|
| > | The right angle bracket, or greater-than sign, indicates menu item selections in a graphic user interface. | File > New > Project | From the File menu, choose New, then from the New submenu, choose Project. |

### Path variable conventions

This table describes the path variable conventions used in this document.

| Path variable | Meaning |
|---|---|
| `$MW_HOME` | Indicates the absolute path to your Oracle Middleware home directory, which is the root directory for your WebLogic installation. |
| `$DOMAIN_HOME` | Indicates the absolute path to your WebLogic domain home directory. For example, if `endeca_server_domain` is the name of your WebLogic domain, then the `$DOMAIN_HOME` value would be the `$MW_HOME/user_projects/domains/endeca_server_domain` directory. |
| `$ENDECA_HOME` | Indicates the absolute path to your Oracle Endeca Server home directory, which is the root directory for your Endeca Server installation. |

## Contacting Oracle Customer Support

Oracle Endeca Customer Support provides registered users with important information regarding Oracle Endeca software, implementation questions, product and solution help, as well as overall news and updates.

You can contact Oracle Endeca Customer Support through Oracle's Support portal, My Oracle Support at *https://support.oracle.com*.

Chapter 1

# Introduction to Endeca Server Security

This section provides overviews of some security topics for both Endeca Server and WebLogic Server.



## Overview of Endeca Server security

The Endeca Server uses SSL for communication between its components. It also contains security filters for queries. The Endeca Server application deployed in the WebLogic domain can be administered by the WebLogic domain administrator.

### Using SSL for encrypted communications

Secure Sockets Layer (SSL) provides secure connections by allowing two applications connecting over a network to authenticate each other's identity and by encrypting the data exchanged between the applications. Authentication allows a server and optionally a client to verify the identity of the application on the other end of a network connection. Encryption makes data transmitted over the network intelligible only to the intended recipient.

Endeca Server allows you to configure mutually-authenticated SSL for communications between its various components. This configuration uses both encryption and certificates for authentication of a component. You can use an Endeca Server utility to create a Certification Authority (CA), which is then used to sign certificates for use by the various components. With two-way SSL (SSL with client authentication), the server presents a certificate to the client and the client presents a certificate to the server. Endeca Server can be configured to require clients to submit valid and trusted certificates before completing the SSL connection.

You quickly and easily create the certificates with the `generate_ssl_keys` utility, which is documented in this guide in *Key Generation Utility on page 18*. After creating the certificates, you can copy them to other Endeca Information Discovery products in your deployment, such as Studio or the Provisioning Service.

The `generate_ssl_keys` utility also enables the SSL Listen Port of 7002 in WebLogic Server, and sets 7002 as the port on which Endeca Server is started. Note that the non-SSL (HTTP) port 7001 is left enabled.

### Query-time security filters for restricting data to end users

The `DataSourceFilter` query filter in Endeca Server allows you to restrict access to sensitive data within your application when end users make data queries. `DataSourceFilter` filters the corpus of records before any other processing is done. In other words, this filter is applied first, and makes the universe of data that is visible to your query smaller. This means that filtered-out records will not contribute to spell correction, and will not be available as part of `AllBaseRecords` in EQL.

Because `DataSourceFilter` restricts the searchable records to a specified subset of the total records in the Dgraph, it can be used as a security filter to prevent users from obtaining records that they are not authorized to view.

After the universe of records has been narrowed by `DataSourceFilter`, the `SelectionFilter` filter can be used for additional application-level filtering. It specifies the criteria for the final record result set. The results that are returned are the records that match all of the filters specified in the query.

By using one or both of these filters, you can effectively control the flow of data from an Endeca data domain to your application front end. For more information on these filters, see the *Oracle Endeca Server Developer's Guide*.

### Administrator role

Unlike in WebLogic, there are no administrator roles in Endeca Server. However, after installing Endeca Server, a WebLogic domain administrator can use the WebLogic Administration Console to stop and start the Endeca Server application.

## Overview of WebLogic Server security

Because the Endeca Server application runs in a J2EE container in the WebLogic Server, you should be aware of some of the security features of WebLogic Server.

### Configuring SSL on WebLogic Server

Oracle recommends that you use SSL in a production environment. WebLogic Server supports SSL on a dedicated listen port which defaults to 7002.

The `generate_ssl_keys` utility not only creates the SSL certificates but also enables the SSL Listen Port of 7002 in WebLogic Server, and sets 7002 as the port on which Endeca Server is started. Therefore, all you need to do to configure SSL on WebLogic Server is to run this script (described in *Key Generation Utility on page 18*).

Note that the `generate_ssl_keys` utility leaves the non-SSL (HTTP) 7001 port enabled, so that you can log into the Administration Console without a certificate.

### Firewalls and connection filters

You should protect your network with a firewall. If you must provide outside access to destinations within the firewall, set up a secure VPN (Virtual Private Network). A firewall limits traffic between two networks. Firewalls can be a combination of software and hardware, including routers and dedicated gateway machines.

WebLogic also supports connection filters for domains. Connection filters allow you to deny access at the network level. They can protect server resources on individual servers, server clusters, or an entire internal network or intranet. For example, you can deny any non-SSL connections originating outside of your corporate network. Network connection filters are a type of firewall in that they can be configured to filter on protocols, IP addresses, and DNS node names. For more information on connection filters, see the "Using Network Connection Filters" chapter in the *Programming Security for Oracle WebLogic Server* document (*http://docs.oracle.com/cd/E23943_01/web.1111/e13711/con_filtr.htm#SCPRG377*).

## Securing the WebLogic Server host

When running your Endeca Server application in a WebLogic Server production environment, it is important that you secure the physical machine, the operating system, and all other software that is installed on the host machine. You should also check with the manufacturer of the machine and operating system for recommended security measures.

The following recommendations for securing a WebLogic Server host in a production environment are summarized from the WebLogic document titled *Securing a Production Environment*, available at the URL listed in the following topic.

| Security Action | Description |
|---|---|
| Physically secure the hardware. | Keep your hardware in a secured area to prevent unauthorized operating system users from tampering with the deployment machine or its network connections. |
| Log out of the Administration Console before navigating to a non-secure site. | If you are logged on to the WebLogic Server Administration Console, be sure to log out completely before browsing to an unknown or non-secure Web site. |
| Secure networking services that the operating system provides. | Have an expert review network services such as e-mail programs or directory services to ensure that a malicious attacker cannot access the operating system or system-level commands. The way you do this depends on the operating system you use.<br>Sharing a file system with other machines in the enterprise network imposes risks of a remote attack on the file system. Be certain that the remote machines and the network are secure before sharing the file systems from the machine that hosts WebLogic Server. |
| Use a file system that can prevent unauthorized access. | Make sure that the file system on each WebLogic Server host can prevent unauthorized access to protected resources. For example, on a Windows computer, use only NTFS. |
| Set file access permissions for data stored on disk. | Set operating system file access permissions to restrict access to data stored on disk. This data includes, but is not limited to, these locations:<br>• The `$DOMAIN_HOME/config/ssl` directory in which the SSL certificates are stored.<br>• The `$DOMAIN_HOME/EndecaServer/data` directory in which the Endeca data domain indexes are stored.<br>• The `$DOMAIN_HOME/EndecaServer/logs` directory in which the Endeca data domain logs are written.<br>For example, Linux provide utilities such as `umask` and `chmod` to set the file access permissions. At a minimum, consider using "umask 066", which denies read and write permission to Group and Others. |

| Security Action | Description |
|---|---|
| Limit the number of user accounts on the host machine. | Avoid creating more user accounts than you need on WebLogic Server host machines, and limit the file access privileges granted to each account. On operating systems that allow more than one system administrator user, the host machine should have two user accounts with system administrator privileges and one user with sufficient privileges to run WebLogic Server. Having two system administrator users provides a backup at all times. The WebLogic Server user should be a restricted user, not a system administrator user. One of the system administrator users can always create a new WebLogic Server user if needed.<br>WebLogic domain and server configuration files should be accessible only by the operating system users who configure or run WebLogic Server.<br>Review active user accounts regularly and when personnel leave. |
| For your system administrator user accounts, choose names that are not obvious. | For additional security, avoid choosing an obvious name such as "system", "admin", or "administrator" for your system administrator user accounts. |
| Safeguard passwords. | The passwords for user accounts on production machines should be difficult to guess and should be guarded carefully. Set a policy to expire passwords periodically.<br>Do not deploy a domain that can be accessed with the username `weblogic` and the password `welcome1`. These credentials are provided by default for domains containing the WebLogic Server sample applications, which should never be installed on a machine in a production environment. |
| Do not develop on a production machine. | Develop first on a development machine and then move code to the production machine when it is completed and tested. This process prevents bugs in the development environment from affecting the security of the production environment. |
| Do not install development or sample software on a production machine. | Do not install development tools on production machines. Keeping development tools off the production machine reduces the leverage intruders have should they get partial access to a WebLogic Server production machine. Do not install the WebLogic Server sample applications on a production machine. |
| Do not run WebLogic Server in development mode in a production environment. | Production mode sets the server to run with settings that are more secure and appropriate for a production environment. |
| Enable security auditing. | Both Linux and Windows support security auditing of file and directory access, Oracle recommends using audit logging to track any denied directory or file access violations. Administrators should ensure that sufficient disk space is available for the audit log. |

| Security Action | Description |
|---|---|
| Consider using additional software to secure your operating system. | Most operating systems can run additional software to secure a production environment. For example, an Intrusion Detection System (IDS) can detect attempts to modify the production environment. Refer to the vendor of your operating system for information about available software. |
| Apply operation-system patch sets and security patches. | Refer to the vendor of your operating system for a list of recommended patch sets and security-related patches. |
| Install the latest maintenance packs, minor releases, and critical patch updates. | If you are responsible for security related issues at your site, register your WebLogic Server installation with My Oracle Support. You can create a My Oracle Support account at *http://www.oracle.com/us/support*.<br>You should also visit the Critical Patch Updates and Security Alerts page at *http://www.oracle.com/technology/deploy/security/alerts.htm*. |
| Secure your JNDI root context. | Group "Everyone" must not have access to the JNDI Root Content resource if the WebLogic Server Administration Console is externally visible. By default, JNDI resources have a default security policy of Everyone. |

## WebLogic Server security guides

Because WebLogic security is an important aspect of your deployment's overall security planning, read the following WebLogic-specific security guides:

- *Understanding Security* (*http://docs.oracle.com/cd/E23943_01/web.1111/e13710/toc.htm*)
- *Securing Oracle WebLogic Server* (*http://docs.oracle.com/cd/E23943_01/web.1111/e13707/toc.htm*)
- *Securing a Production Environment* (*http://docs.oracle.com/cd/E23943_01/web.1111/e13705/toc.htm*)
- *Securing Resources Using Roles and Policies* (*http://docs.oracle.com/cd/E23943_01/web.1111/e13747/toc.htm*)

Chapter 2

# Installation and OS Security

This section provides information on an installation strategy and security resources for your Linux or Windows operating system.

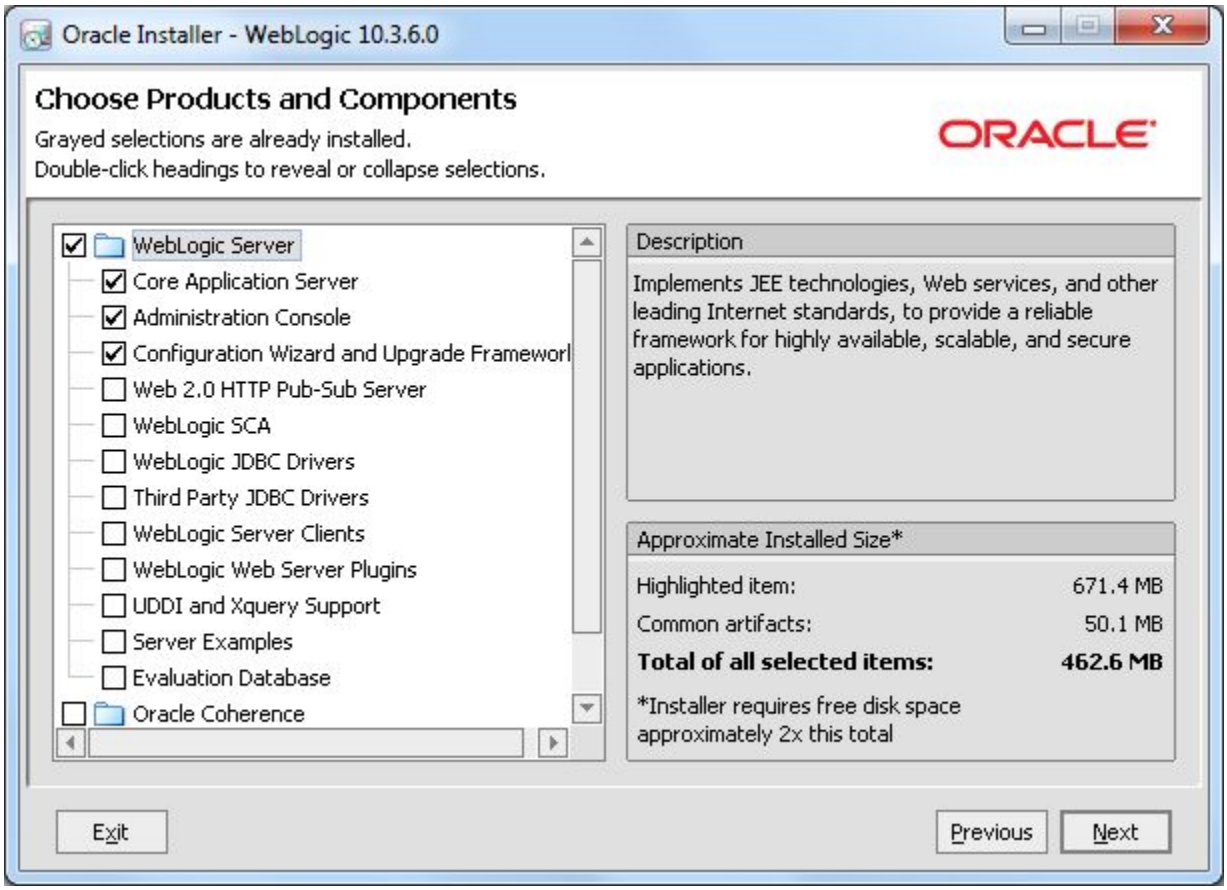


## Installing WebLogic securely

Minimizing your software footprint contributes to a security system because you are reducing the possible areas of attack.

To reduce your software footprint, you should install only those Oracle and third-party software packages that you need to deploy your application.

The WebLogic Server installation that is recommended in the *Oracle Endeca Server Installation Guide* installs the minimum number of packages that are required by an Endeca Server deployment. The **Choose Products and Components screen** of the installer looks like this:




As the screen shot shows, only the Core Application Server, Administration Console, and Configuration Wizard are installed. Note that the Oracle Coherence product was not installed because Endeca Server does not utilize it.

In particular, you should not install the Server Examples package, which includes the Derby database used by the sample applications.

Do not run WebLogic Server when configured in development mode. Instead, make sure WebLogic Server is configured to run in production mode. Production mode sets the server to run with settings that are more secure and appropriate for a production environment.

# Hardening the Linux OS

If you use the Linux operating system, you should read two OTN (Oracle Technology Network) articles on security, as well as an NSA security document.

## OTN articles

The first OTN article is titled *"Tips for Hardening an Oracle Linux Server"* and is available at this URL: *http://www.oracle.com/technetwork/articles/servers-storage-admin/tips-harden-oracle-linux-1695888.html*. The article provides tips and techniques for hardening an Oracle Linux server, covering the following topics:

- Minimize software and services. Eliminating unnecessary software packages and services minimizes possible avenues of attack.
- Tighten networking and user access. The network is a prime point of entry for malicious users and applications. Fine-tuning the network configuration, along with all user access points, helps to prevent unauthorized access.
- Protect applications and data. Setting up devices, mounts, and file systems appropriately (and in some cases using encryption) helps to safeguard applications and data.
- Implement security features that enforce policies. In some cases, the security policy may dictate additional mechanisms, such as TCP wrappers, Pluggable Authentication Modules (PAM), or the implementation of Security-Enhanced Linux (SELinux).

The second OTN article is titled *"Tips for Securing an Oracle Linux Environment"* and is available at this URL: *http://www.oracle.com/technetwork/articles/servers-storage-admin/secure-linux-env-1841089.html*. The article provides software, network, and system monitoring recommendations for maintaining a secure Oracle Linux environment. The strategies discussed are:

- Maintain physical system security.
- Use security and management tools to scan for signs of compromise.
- Apply software and security updates promptly.
- Review procedures and policies at regular intervals.

## NSA documents

The National Security Agency publishes a document titled *"Guide to the Secure Configuration of Red Hat Enterprise Linux 5"*, available at: *http://www.nsa.gov/ia/_files/os/redhat/NSA_RHEL_5_GUIDE_v4.2.pdf*

The NSA also publishes a short pamphlet on hardening tips for Red Hat Enterprise Linux 5, which is available at: *http://www.nsa.gov/ia/_files/factsheets/rhel5-pamphlet-i731.pdf*

# Windows security resources

The Microsoft TechNet site is a starting point for Windows security resources.

The Windows Server Security page provides security guidance topics for the Windows Server operating system in general. The page is located at: *http://technet.microsoft.com/en-us/windowsserver/ff843381*

The Secure Windows Server page provides security topics that are specific to the Windows Server 2008 operating system. The page is located at: *http://technet.microsoft.com/library/dd548350%28WS.10%29.aspx*

For example, one of the security resources from Secure Windows Server page is the *Threats and Countermeasures Guide*.

You can use the Microsoft Security Compliance Manager, which includes extensive guidelines and documentation about hardening the server. You can download the manager from: *http://www.microsoft.com/en-us/download/details.aspx?id=16776*

Chapter 3

# Endeca Server Communications

This section discusses communications and authentications between Endeca Server and its client programs.



## Endeca Server authentication of clients

The Endeca Server application running in the WebLogic domain uses SSL mutual authentication when it communicates with the Dgraph process and the `endeca-cmd` command utility.

Mutually-authenticated SSL uses both encryption and certificates for authentication of a component. Thus, safe and trusted communication among the various components of your Endeca Server application ensures that the data being transmitted will not be compromised.

During communication initialization, each component confirms that the certificate it is receiving from the other party has been signed by the CA. A bearer's possession of a signed certificate implicitly grants full access to the Endeca Server component it is contacting. You create the SSL certificates for these components with the `generate_ssl_keys` utility, which is documented in *Key Generation Utility on page 18*.

Since Endeca Server runs in the WebLogic Server container, it supports both the SSL V3.0 and TLS V1.0 protocols, as the WebLogic Server itself. (Earlier versions of these protocols are not supported in the Endeca Server).

## Endeca Server and the endeca-cmd command interface

In a secure installation of the Endeca Server, the `endeca-cmd` utility communicates with the Endeca Server securely.

Two versions of the `endeca-cmd` command utility exist — SSL-enabled and non-SSL. The SSL version of `endeca-cmd`is located in the `$DOMAIN_HOME/EndecaServer/bin` directory, and is the one you should be using for an Endeca Server SSL deployment.

The SSL version lets you issue `endeca-cmd` commands without having to specify the location of the SSL certificates with the `--keystore` and `--truststore` options (as they are used automatically). It does so by making use of the `$DOMAIN_HOME/config/EndecaCmd.properties` configuration file. The file paths in `EndecaCmd.properties` are set automatically by the `generate_ssl_keys` utility.

When you run the `endeca-cmd` command utility in a secure installation, you need to specify the location of the Java keystore file (`endecaServerClientCert.ks`) and the Java truststore file (`endecaServerTrustStore.ks`), as well as the password for the key and trust stores. Providing this information enables the `endeca-cmd` command utility to communicate securely with the Endeca Server.

By default, you are prompted for the keystore password whenever you issue a command. This is the password that you specified with the `--sslPassphrase` flag of the `generate_ssl_keys` utility. Note that you can override the prompt by providing the password with the `--password` option; however, you should not create a script that uses the `--password` option, as the password would have to be specified in clear text.

If Endeca Server receives an HTTP request while in HTTPS-only mode, the request is denied with this error message:

```
SEVERE: Error while invoking endpoint "http://localhost:7001/endeca-server/ws/manage" from client
endeca-cmd encountered a problem.

caused by:
Error contacting the Endeca Server localhost:7001: Client received SOAP Fault from server :
OES-000149: Only allowing https connections, received http

caused by:
Client received SOAP Fault from server : OES-000149: Only allowing https connections, received http
```

For more information on the SSL version of the `endeca-cmd` command utility, see the *Oracle Endeca Server Administrator's Guide*.

# Communication between Endeca Server and the Dgraph process

When SSL is enabled on the Endeca Server, it is also enabled on the Dgraph process. Additionally, it is recommended that all clients of the Endeca Server contact the particular data domain (and any of its Dgraph processes) through the Endeca Server Java application hosted by the WebLogic Server.

The following statements describe how the communication between the Endeca Server and the Dgraph process is handled:

- **SSL and the Dgraph process**.
  If you install Endeca Server with SSL (recommended), then when you create a data domain in the Endeca Server, you do not have to manually specify the SSL certificate files for the Dgraph process as the Endeca Server automatically performs that configuration task. The `generate_ssl_keys` utility updates the `$DOMAIN_HOME/config/EndecaServer.properties` file with the pathnames of the SSL certificate file (`dgraphCert.pem`) and certificate authority file (`dgraphCA.pem`). When you create a data domain in the Endeca Server, the Dgraph process associated with that data domain is automatically configured to run with both of these files via the Dgraph `--sslcertfile` and `--sslcafile` flags. Thereafter, these certificates are used for authentication between Endeca Server and the Dgraph process.
- **Hostname of the Dgraph process.** The Endeca Server and its Dgraph processes are always assumed to be running on the same machine. All the URLs reference `localhost`. When HTTPS is used between the Endeca Server and the Dgraph processes it is controlling, The Endeca Server does not explicitly validate

the hostname. Therefore, the Endeca Server is hard-coded not to check that the hostname in the Dgraph's certificate matches the URL that it is talking to.

- **Communication with the Dgraph process**. It is recommended that all clients establish a connection to the Dgraph through the Endeca Server Java application. If, as a system administrator, you must communicate with the Dgraph directly (this is typically not necessary), you should be required to first log into the physical machine hosting the Dgraph.

# About connecting Web browsers to data domains

You should never allow user Web browsers to connect directly to the machine hosting the Endeca Server and the Endeca data domains. Browsers started by non-administrators should always connect to your application through an application server. If you use Studio with the Endeca Server, this requirement is satisfied by user authentication and security features in Studio.

# IPv4 and IPv6 address support

The Oracle Endeca Server and its Dgraph process support both IPv4 (Internet Protocol Version 4) and IPv6 (Internet Protocol Version 6) addressing schemes for connections.

This IPv4 and IPv6 addressing support is configured automatically in the Oracle Endeca Server and the Dgraph, so there is no need for the administrator to do any explicit addressing configuration.

# Encryption

When installed securely over SSL, the Endeca Server supports SSL/TLS ciphers for encryption of its messages between the Endeca Server and the Dgraph.

In particular, the following ciphers and cipher methods are disabled: EXPORT ciphers, ciphers with no authentication (aNULL) and no encryption (eNULL), MD5 hash functions, and RC4.

# Cluster Coordinator authentication

The Dgraph process and Endeca Server both rely on structures in the Cluster Coordinator for proper operation.

In order to prevent a malicious or malfunctioning element in the environment from affecting these structures, the programs using Cluster Coordinator are authenticated by the Coordinator before reading or writing these structures.

## ACLs

The Cluster Coordinator uses ACLs to control access to the data nodes of the Cluster Coordinator data tree. The ACL implementation is similar to Linux file access permissions: it employs permission bits to allow/disallow various operations against a node and the scope to which the bits apply. However, unlike standard Linux permissions, a Cluster Coordinator node is not limited by the three standard scopes for user, group, and world. Instead, an ACL specifies sets of IDs and permissions that are associated with those IDs.

The IDs and permissions are specified via the Cluster Coordinator's built-in digest authentication scheme. The digest authentication requires the client to provide a name and password, and likewise allows access to users authenticated with particular names and passwords.

### Dgraph process and Endeca Server ACLs

The Dgraph process and Endeca Server both use a single cluster-wide name and password. The name is always "endeca", but the password is randomly generated for each cluster deployment. Immediately upon establishing a session with the Cluster Coordinator, the client authenticates with these credentials. Then, whenever creating a Cluster Coordinator node, it attaches an ACL requiring these credentials for any access.

### Credential storage

The password for Cluster Coordinator access is stored in the WebLogic Credential Store Framework (CSF), the standard secret store for Oracle Fusion Middleware products. The basic interface for CSF is in the Java libraries provided as part of a WebLogic deployment. Therefore, the Endeca Server can retrieve the Cluster Coordinator credentials from the CSF. For the Dgraph process usage, the Endeca Server gets the Cluster Coordinator credentials from the CSF and passes them to the Dgraph process.

## Security considerations for the Endeca Server Cluster

You should be aware of additional security concerns if you have an Endeca Server cluster deployment instead of a single-machine Endeca Server deployment.

An Endeca Server cluster is a deployment of a cluster of multiple Endeca Server instances that host and manage multiple data domains. Details on this type of environment are provided in the *Oracle Endeca Server Cluster Guide*.

### Shared file system permissions

One major security consideration in an Endeca Server cluster is that all Dgraph nodes in the same data domain utilize the same index files residing on a shared storage drive. In addition, the Endeca Server cluster state is also maintained in the shared file system by the Cluster Coordinator.

These files should be owned by the user ID that is running the WebLogic process. That user ID should have full access to the directory and files, while everyone else can have no access. For example, on a Linux system, you can set permissions on the topmost directory to `700`.

### Multi-machine security

Because the Endeca Server cluster will run across multiple machines, you should ensure that communications channels between the machines are secure, and that each machine itself is secure.

Chapter 4

# Key Generation Utility

This section describes the utility used to generate SSL certificates.



## Key generation utility syntax

The `generate_ssl_keys` utility creates the SSL certificate keys.

There are specific versions of the `generate_ssl_keys` utility for each operating system:

- Linux: `generate_ssl_keys.sh`
- Windows: `generate_ssl_keys.bat`

The utility is located in the `$DOMAIN_HOME/EndecaServer/bin` directory. For example, if `endeca_server_domain` is the name of your WebLogic domain for Endeca Server, then the default path on Windows is:

```
C:\Oracle\Middleware\user_projects\domains\endeca_server_domain\EndecaServer\bin
```

**Important:** If you are deploying an Endeca Server cluster, make sure to generate SSL certificates after you have installed the Endeca Server on the Admin Server and before you have cloned the Admin Server to create Managed Servers.

The syntax for the utility is:

```
generate_ssl_keys --username <wls-domain-admin-username> --password <wls-domain-password>
     --sslPassphrase <phrase> [--url <wls-admin-url>] [--syncOnly]
```

The meanings of the flags are:

| Flag | Meaning |
|---|---|
| `--username` | Mandatory. Specifies an admin username for this domain. You can use the same username that you specified when you created the WebLogic domain for the Endeca Server application. |
| `--password` | Mandatory. Specifies the password for the username. You can use the same password that you specified when you created the username for the WebLogic domain for the Endeca Server application. |

| Flag | Meaning |
|---|---|
| `--sslPassphrase` | Mandatory. Specifies the passphrase for the new SSL keys. |
| `--url` | Optional. Specifies the URL of the WebLogic Server if it is running on a host:port other than the default. It defaults to<br>`t3://localhost:<server_port>`<br>(where *server_port* is the port you specified in the installer). The script runs against `localhost` and *server_port*. If either or both have changed, use this flag to specify the correct host:port. Note that the argument must be a full URL and it must use the `t3` protocol. |
| `--syncOnly` | Optional. This flag re-synchronizes your existing SSL keys across your Endeca Server cluster deployed in the WebLogic domain (that is, across a set of Managed Servers). This flag does not take an argument. Running the `generate_ssl_keys` utility with `--syncOnly` processes the keys for each of the existing Managed Servers and the Admin Server to make sure they all are set with the same SSL configuration.<br>Using this flag is useful, for example, if you deploy a brand-new Managed Server in the Endeca Server cluster and do not set up the SSL configuration in it properly.<br>Note that you cannot change the passphrase with this flag; this means that you must specify an existing passphrase with the `--sslPassphrase` flag when you use the `--syncOnly` flag. |

### Expiration date

The server and client certificates are valid for 1460 days (4 years) from the time that they are generated. When they expire, you must generate new keys.

### Usage example

A usage example is:

```
generate_ssl_keys --username ESUser --password welcome1 --sslPassphrase thx1138
```

# Creating SSL certificates

This topic describes how to run the `generate_ssl_keys` utility.

Before running this utility, make sure that you have chosen a strong passphrase for the keys. For detailed information on the utility's syntax, see *Key generation utility syntax on page 19*.

**Important:** If you are deploying an Endeca Server cluster, make sure to generate SSL certificates after you have installed the Endeca Server on the Admin Server and before you have cloned the Admin Server to create Managed Servers in the WebLogic domain configured for the Endeca Server application.

To generate SSL certificates:

1. Start the Admin Server for the Endeca Server domain in WebLogic.

   The start-up procedure should ask you for the administrator user name and password that you specified when you created the WebLogic domain.

2. From a command prompt, change to the `$DOMAIN_HOME/EndecaServer/bin` directory.

3. Run the `generate_ssl_keys` utility with a domain username/password and specify the passphrase for the certificates. For example:

   ```
   generate_ssl_keys --username ESUser --password welcome1 --sslPassphrase thx1138
   ```

   A successful procedure is indicated when you see a message that ends as follows:

   ```
   The following non-dynamic attribute(s) have been changed on MBeans
   that require server re-start:
   MBean Changed : com.bea:Name=AdminServer,Type=SSL,Server=AdminServer
   Attributes changed : HostnameVerificationIgnored, JSSEEnabled

   Activation completed

   Done! Your WLS server(s) may need to be restarted for
   all changes to take effect.
   ```

4. Stop and then re-start the WebLogic Admin Server.

The `generate_ssl_keys` utility creates these SSL certificates in the `$DOMAIN_HOME/config/ssl` directory:

- `dgraphCA.pem` — Certificate authority file used by all clients and servers to authenticate the other endpoint of a communication channel with the Endeca Server. Used with the Dgraph `--sslcafile` flag.
- `dgraphCert.pem` — Certificate file used by all clients and servers to specify their identity when using SSL to connect to the Oracle Endeca Server. This certificate should be thought of as the identity of the system powered by the Dgraph, or as the identity of all components of the system. Used with the Dgraph `--sslcertfile` flag.
- `endecaServerCerts.ks` — Java identity keystore.
- `endecaServerClientCert.ks` — Java keystore used for Endeca Server clients. Used for the `keystore` parameter of `EndecaCmd.properties`.
- `endecaServerTrustStore.ks` — Java truststore used for Endeca Server clients. Used for the `truststore` parameter of `EndecaCmd.properties`.
- `esClientCert.p12` — Personal Information Exchange (PKCS12-format) key file. Note that this client key has its own password, which is the user-entered passphrase plus "clientkey" appended.

Besides generating the SSL keys, the utility also:

- Updates the `EndecaServer.properties` and `EndecaCmd.properties` files (in the `$DOMAIN_HOME/config` directory) with the pathnames of the key files.
- Enables the SSL Listen Port of 7002 in WebLogic Server, and sets 7002 as the port on which Endeca Server is started.
- For the Admin Server, sets `endecaServerCerts.ks` as the custom identity keystore and `endecaServerTrustStore.ks` as the custom trust keystore. Both settings are visible from the Admin Server's Keystores tab.

- For the Admin Server, sets `Oracle Endeca Server Certificate` as the Private Key Alias. This setting is visible from the Admin Server's SSL tab.

Note that although the SSL port 7002 is enabled, the non-SSL (HTTP) port 7001 is still enabled.

# Regenerating keys

You can regenerate your SSL keys with a new passphrase.

This task presumes that you have already generated a set of SSL keys and now want to regenerate them with a different passphrase.

When generating key certificates, keep in mind that you cannot generate a new set of keys if a previous set already exists. If you attempt to do so, the `generate_ssl_keys` utility fails with this error message:

```
SSL key files already exist!

If you intend to re-push existing SSL config across your WLS cluster,
run again and add the --syncOnly flag

Exiting with error state!
```

In this case, you must use the following procedure.

To regenerate the SSL key certificates:

1. Go to the `$DOMAIN_HOME/<endeca_server_domain>/config/ssl` directory (where *endeca_server_domain* is the name of your WebLogic domain for the Endeca Server).

   You can use a command prompt or Windows Explorer (on Windows).
2. Delete all the certificate key files.
3. Run the `generate_ssl_keys` utility, as documented in *Creating SSL certificates on page 20*.

   You will be specifying the new passphrase with the `--sslPassphrase` flag.
4. Stop and then restart the WebLogic Admin Server.

After the server restarts, you will use the new passphrase when you run Endeca Server commands.

# Index