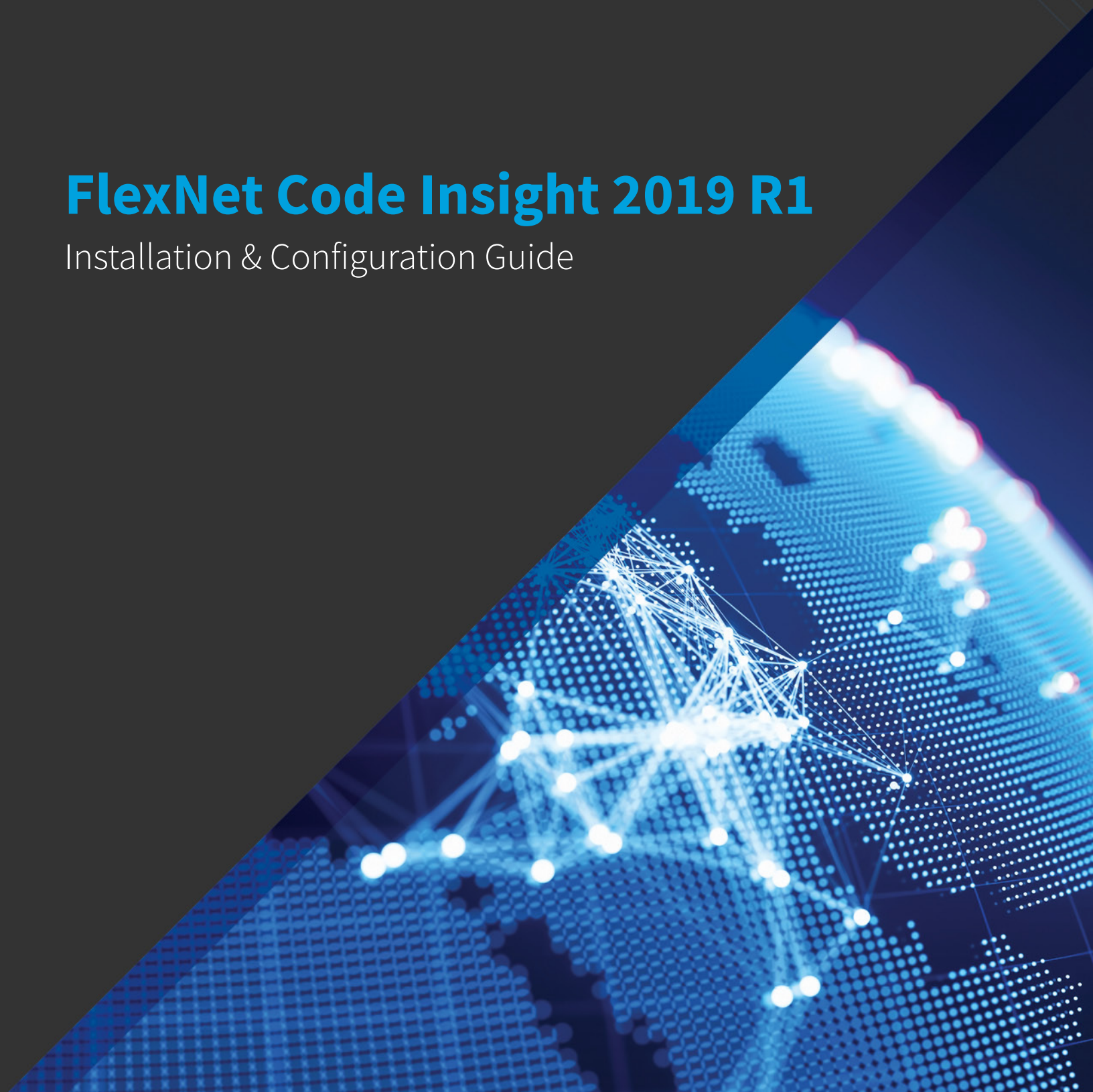
Flexera

FlexNet Code Insight 2019 R1

Installation & Configuration Guide



Legal Information

Book Name: FlexNet Code Insight 2019 R1 Installation & Configuration Guide

Part Number: FNCI-2019R1-IG00

Product Release Date: March 2019

Copyright Notice

Intellectual Property

Restricted Rights Legend

Contents

1

Installing FlexNet Code Insight

This section contains the following topics covering the installation and startup of FlexNet Code Insight:

System Requirements

Before installing FlexNet Code Insight, ensure that the following requirements are addressed for your system:

- A supported database instance and its associated connector. See Database Support for a description of supported databases and connectors.
- A FlexNet Code Insight license key file (`codeinsight.key`)
- On Linux machines, ensure that the number of open file handles is greater than 50k, a value typically set with the `ulimit` command. For more information about the open file limit, see Setting the Open File Limit for Linux/Unix.

Important • *This requirement for the open file limit is absolutely essential for FlexNet Code Insight to function properly on Unix and Linux platforms.*

- Any requirements specific to your FlexNet Code Insight plugin and remote data source. Refer to the *FlexNet Code Insight Plugins Guide* for details.

Note • *The JRE is included in the installation; a separate download is not necessary. Only JRE 8 is supported.*

The following provides additional requirements:

- Platform Support
- Database Support
- Browser Support
- Recommended Hardware
- Recommended Software

Platform Support

FlexNet Code Insight supports the following platforms:

- Windows Server 2012
- Windows Server 2016
- RHEL 6.x, 7.x
- CentOS 6.x, 7.x

Database Support

FlexNet Code Insight requires that either a MySQL or SQL Server database be installed. The following lists components required to install and configure a database for use by Code Insight:

- MySQL Required Components
- SQL Server Required Components

MySQL Required Components

The following describes the components needed to install and run MySQL as the FlexNet Code Insight database:

- The community edition of MySQL 8.0 (also known as MySQL 5.8) or MySQL 5.7, downloaded from https://dev.mysql.com/downloads/mysql.

  Note • *Code Insight does not support the Docker version of My SQL. (It supports the native version only.)*

- The appropriate JDBC driver connector file:
  - For MySQL 8.0, use the latest version of `mysql-connector-java-8.0.x.jar`. You can download this file from https://dev.mysql.com/downloads/connector/j/.

- For MySQL 5.7, use `mysql-connector-java-5.1.x-bin.jar`. You can download this file from http://dev.mysql.com/downloads/connector/j/5.1.html.

  The connector is required to enable FlexNet Code Insight to connect to the MySQL database.

  This file must reside in your `tomcat/lib` folder. (The Code Insight installer will automatically copy the file to this location.)

- An environment that can support the required size settings listed in Required MySQL Database Settings.
- A database instance configured with the settings described in Required MySQL Database Settings and Binary Logging Option for MySQL.

SQL Server Required Components

The following lists the required components needed to install and run SQL Server as the Code Insight database:

- SQL Server 2016 Sp2 (recommended version for best performance).
- The JDBC driver connector file, `mssql-jdbc-6.4.0.jre8.jar`, which you can download from https://www.microsoft.com/en-us/download/details.aspx?id=56615. Copy this file to your `tomcat/lib` folder.

  The connector is required to enable Code Insight to connect to the SQL Server database.

- The package `sql_server_pre_install_scripts.zip` containing the scripts needed to set up the SQL Server database for Code Insight. See Downloading the Scripts Needed to Set Up the SQL Server Database for instructions on the download process.
- At least one disk (OS or non-OS) with 100 GB free space.

Downloading the Scripts Needed to Set Up the SQL Server Database

Use the following steps to download the package containing the script files needed to set up the SQL Server database for Code Insight.

Task ***To download the package containing the scripts, do the following:***

1. Log into the Customer Community page of the Flexera website:

   `https://flexeracommunity.force.com/customer/`

2. Click **Downloads**.
3. Click the **Access** button under **FlexNet Code Insight**. The Product and License Center page appears.
4. Select **FlexNet Code Insight** from the **Your Downloads** list.
5. Select the version of FlexNet Code Insight from the list. The **Downloads** page appears.
6. Download the `sql_server_pre_install_scripts.zip` file.
7. When the download finishes, extract the following files to a location accessible for later execution using the SQL Server console, as described in Setting Up the SQL Server Database:

   - `codeinsight_serversettings.sql`
   - `codeinsight_db_creation_with_maintainenceplan.sql`

A third script, `codeinsight_db_drop_with_maintainenceplan.sql`, is used to drop the database and is *not* used as part of the database setup. Instructions for dropping the database are found in Dropping the SQL Server Database.

Browser Support

FlexNet Code Insight supports the following browsers:

- Chrome (latest stable version)
- Internet Explorer (latest stable version)
- Firefox (latest stable version)

Note • *FlexNet Code Insight no longer allows uppercase or mixed case when entering the application's URL. To start FlexNet Code Insight in a browser, you must enter* ***codeinsight*** *in lowercase.*

Recommended Hardware

The recommended deployments and configurations are explained in this section:

- Deployment Models
- Configuration Guidelines

Deployment Models

The FlexNet Code Insight deployment model may be configured as a single-node. Each deployment consists of the following elements:

Table 1-1 • Deployment Models

Entity	Description
Core Server	Main interface to FlexNet Code Insight.
Scan Server	Contains codebase to be scanned (required for local scans only, not for remote scans).
Database	Central database containing all library metadata supplied by electronic update and all stored scan results.
Compliance Library (CL) (Optional)	Library containing all the data required to perform source-code fingerprint (snippet) matching and exact-file matching. The Scan Server must have access to the CL through a mapped or mounted drive.

Note • *A multi-scan deployment model is not available in this release.*

Configuration Guidelines

The following table shows supported configurations for the various FlexNet Code Insight entities. For optimal performance, use these configuration guidelines:

- Use the Single Server Configuration (see below), in which the Core Server, Scan Server, Database, and Compliance Library (CL) are installed on the same machine.
- When installing the CL on the same machine as the Scan Server, install it on a drive or volume different from the one on which the Scan Server resides.

Table 1-2 • Supported Configurations

Configuration	CPU (Cores)	Memory	Disk Space
Single Server (*highly recommended*): Core Server Scan Server Database Compliance Library (CL)	2-CPU (each at least 2 GHZ+) with 8+ cores on the server	64 GB	**Server:** 500 GB High-speed Disk for the Database (SSD Recommended) 500 GB High-speed Disk for the Core/Scanner to store the codebase 950 GB High-speed Disk for the CL
Server 1: Core/Scanner/ Compliance Library (CL) **Server 2**: Database	2-CPU (each at least 2 GHZ+) 8+ cores on each server	**Server 1**: 32 GB **Server 2**: 32 GB	**Server 1:** 500 GB High-speed Disk for Core/Scanner to store the codebase 950 GB High-speed Disk for the CL **Server 2**: 500 GB High-speed Disk for the Database (SSD Recommended)
Server 1: Core **Server 2**: Scanner/ Compliance Library (CL) **Server 3**: Database	2-CPU (each at least 2 GHZ+) 8+ cores on each server	**Server 1**: 32 GB **Server 2**: 32 GB **Server 3**: 32GB	**Server1:** 250GB High-speed Disk for Core **Server2:** 500 GB High-speed Disk for Scanner to store the codebase 950 GB High-speed Disk for the CL **Server 3:** 500 GB High-speed Disk for the Database (SSD Recommended)

Recommended Software

The following software is recommended for FlexNet Code Insight.

Database Client

A SQL client or command-line interface is necessary to run database scripts. The following free SQL clients are available:

- HeidiSQL: http://www.heidisql.com/download.php
- MySQL Workbench: http://www.mysql.com/products/workbench/

Preparing to Install FlexNet Code Insight

Installing FlexNet Code Insight is a simple, prompt-driven process, but before beginning the installation, you will need to do the following:

- Ensure that you have met the prerequisites in System Requirements.
- Follow the procedure in Setting Up the Database.
- Perform any additional environmental and communication configuration for Code Insight, such as the following:
  - Network and Firewall Considerations
  - Setting the Open File Limit for Linux/Unix
  - Enabling Secure HTTP Over SSL
  - Configuring a Networking Proxy Server Connection

Setting Up the Database

Before you install FlexNet Code Insight, a database administrator must set up the MySQL or SQL Server database for use by Code Insight:

- Setting Up the MySQL Database
- Setting Up the SQL Server Database

Setting Up the MySQL Database

The following topics describe how configure the MySQL database for FlexNet Code Insight:

- Setting Up a MySQL Instance
- Required MySQL Database Settings
- Binary Logging Option for MySQL
- Sample Procedure for Creating an Appropriate Database Schema and User

Setting Up a MySQL Instance

The database administrator needs to perform the following steps to set up the MySQL database for FlexNet Code Insight.

Task ***To set up the MySQL database for Code Insight, do the following:***

1. Install the MySQL instance, configuring the instance as described in Required MySQL Database Settings and Binary Logging Option for MySQL.

   You might need to configure certain settings once the installation is complete.

   Note • *Installing the instance on a server other than the one on which Code Insight is installed might cause performance degradation.*

2. Create a database schema (with a recommended name of *codeinsight*) and a user who has appropriate access privileges to access the database. The procedure described in Sample Procedure for Creating an Appropriate Database Schema and User can be used to perform these tasks.

Required MySQL Database Settings

FlexNet Code Insight requires the MySQL database configuration described in this table to ensure best performance.

Table 1-3 • Required MySQL Database Settings

Property	System Variable	Recommended Value
Storage Engine	default-storage-engine	innodb
Character Set/Collation	character-set-server/ collation-server	utf8mb4/ utf8mb4_unicode_ci
InnoDB Buffer Pool Size	innodb_buffer_pool_size	12GB
InnoDB Log File Size	innodb_log_file_size	8GB
Maximum Allowed Packets	max_allowed_packet	100MB

If you need to verify the current settings in your MySQL installation, click the appropriate **Property** link in the table for a description of the verification command. If you need to change a setting in your installation, use the following procedure.

Task ***To configure the MySQL database, do the following:***

1. Within your MySQL installation, do one of the following:

   - As root user in Linux, open the `my.cnf` file (typically located in `/etc/`).
   - As Administrator in Windows, open `my.ini` file (typically located in `C:\ProgramData\MySQL\MySQL Server version`).

2. Edit the settings as shown in the previous table. (If necessary, click the appropriate **Property** link in the table for a description of how to configure a given setting.)
3. After you have updated the settings, restart the database server.

Storage Engine

Select InnoDB as the storage engine. In MySQL, InnodDB is the default engine, so a change is unlikely to be necessary.

To verify the current storage engine, use the following command:

```
SELECT * FROM INFORMATION_SCHEMA.ENGINES;
```

To change the default storage engine, use the appropriate procedure:

- In Linux, edit or add the following line in the [mysqld] section of the my.cnf file:

  ```
  default-storage-engine=innodb
  ```

- In Windows, edit or add the following line in the [mysqld] section of the my.ini file:

  ```
  default-storage-engine=innodb
  ```

Character Set/Collation

Select utf8mb4 as the character set when installing the FlexNet Code Insight MySQL database server.

To verify the current character set and collation, use the following command:

```
SELECT @@character_set_database, @@collation_database;
```

To change the character set and collation, use the appropriate procedure:

- In Linux, edit or add the following line in the [mysqld] section of the my.cnf file:

  ```
  character-set-server=utf8mb4
  collation-server=utf8mb4_unicode_ci
  ```

- In Windows, edit or add the following line in the[mysqld] section of the my.ini file:

  ```
  character-set-server=utf8mb4
  collation-server=utf8mb4_unicode_ci
  ```

InnoDB Buffer Pool Size

Set the InnodDB buffer pool size to at least 12GB.

To verify the current InnoDB buffer pool setting, use the following command. (The resulting value is in GBs.)

```
SELECT @@innodb_buffer_pool_size/1024/1024/1024;
```

To change the InnoDB buffer pool size, use the appropriate procedure:

- In Linux, edit or add the following line in the [mysqld] section of the my.cnf file:

  ```
  innodb_buffer_pool_size=12G
  ```

- In Windows, edit or add the following line in the [mysqld] section of the my.ini file:

  ```
  [mysqld]
  innodb_buffer_pool_size=12G
  ```

InnoDB Log File Size

Set the InnoDB log file size to at least 8GB.

To verify the current InnoDB log file size, use the following command:

```
show variables like 'innodb_log_file_size%';
```

To change the InnoDB log file size, use the appropriate procedure:

- In Linux, edit or add the following line in the `[mysqld]` section of the `my.cnf` file:

  ```
  innodb_log_file_size=8G
  ```

- In Windows, edit or add the following line in the `[mysqld]` section of the `my.ini` file:

  ```
  innodb_log_file_size=8G
  ```

Maximum Allowed Packets

Set the maximum packet size to 100MB.

To verify the current maximum packet size, use the following command:

```
SHOW VARIABLES LIKE 'max_allowed_packet';
```

To change the maximum packet size, use the appropriate procedure:

- In Linux, edit or add the following line in the [mysqld] section of the `my.cnf` file:

  ```
  max_allowed_packet=100M
  ```

- In Windows, edit or add the following line in the `[mysqld]` section of the `my.ini` file:

  ```
  max_allowed_packet=100M
  ```

Binary Logging Option for MySQL

MySQL offers Binary Logging, an advanced feature that captures changes between backups and stores this information in binary log files. The log files—containing information about each statement that modified (or might have modified) the database and the amount of time it took to make the change—are mainly used for data recovery and replication efforts. The files reside in either location:

- The `/var/lib/mysql` directory on Linux
- The program data directory on Windows (for example, `C\ProgramData\MySQL\MySQL Server 8.0\Data`)

Possible Issues with Binary Logging

Binary Logging can cause issues when a Code Insight Electronic Update or scan is run. During either of these processes, the database can be updated with a significant number of insert, update, and delete events. With Binary Logging enabled, details for each event are also written to rolling binary log files, with each file being about 1 GB in size. If these log files are not purged regularly, out-of-memory issues will occur.

The user can decide whether Binary Logging should be enabled.

Disabling or Enabling Binary Logging

By default, Binary Logging is enabled in MySQL8, but disabled in MySQL5.

Task ***To disable or enable Binary Logging, do the following:***

1. Within your MySQL installation, do one of the following:
   - As root user in Linux, open the `my.cnf` file (typically located in `/etc/`).
   - As Administrator in Windows, open `my.ini` file (typically located in `C:\ProgramData\MySQL\MySQL Server` *`version`*).
2. To enable Binary Logging, add or uncomment the `log-bin` line:

   ```
   #Binary Logging
   log-bin="<binaryLogBaseName>"
   ```

   or

   To disable Binary Logging, comment-out the `log-bin` line:

   ```
   #Binary Logging
   #log-bin="<binaryLogBaseName>"
   ```
3. Restart the database server.

Sample Procedure for Creating an Appropriate Database Schema and User

The following is a sample procedure that the database administrator can to create a Code Insight database schema and a database user.

Task ***To create a database schema and user, do the following:***

1. At the command line, log into MySQL as the root user:

   ```
   mysql -u root -p
   ```
2. Type the MySQL root password, and press **Enter**.
3. To create a database and user, type the following command, replacing the username (*fnciuser)* with the user you want to create, and replace *Fnci%1234* with the user's password:

   ```
   CREATE DATABASE codeinsight;
   CREATE USER fnciuser IDENTIFIED BY 'Fnci%1234';
   GRANT ALL ON codeinsight.* TO 'fnciuser'@'%';
   ```
4. Provide the user name and password and the database schema to the person who will install Code Insight.

Setting Up the SQL Server Database

Setting up the SQL Server database for Code Insight involves two phases:

- Phase 1: Install the SQL Server Instance
- Phase 2: Set Up the SQL Server Database

The DBA performs these steps.

Phase 1: Install the SQL Server Instance

Task ***To install the SQL Server instance, do the following:***

1. Install the SQL Server instance, following the instructions included with the SQL Server installer. During the installation, select the appropriate options that do the following:
   - Set the character set (or collation) is to `SQL_Latin1_General_CP1_CI_AS`.
   - Enable the SQL Server Agent.
2. When the installation is complete, start up the SQL Server Agent using the instructions provided in the SQL Server documentation. This a required step for setting up the SQL Server database, described in the next section, Phase 2: Set Up the SQL Server Database.

Phase 2: Set Up the SQL Server Database

Once you have installed the SQL Server instance and have started up the SQL Server Agent, use the following instructions to set up the SQL Server database for Code Insight.

Task ***To set up the SQL Server database for Code Insight, do the following:***

1. Ensure that you have downloaded and extracted the required the Code Insight scripts, as described in Downloading the Scripts Needed to Set Up the SQL Server Database.
2. Understand the purpose of the scripts before executing them:
   - **codeinsight_serversettings.sql**—This script configures the database server to enable the maximum performance for Code Insight. The script sets the following server parameters:
     - **Cost of parallelism**—15 (the threshold at which the optimizer chooses parallel processing)
     - **Max degree of parallelism**—Number of threads created specifically for this configuration.
     - **Max memory configuration**—The server's maximum utilization (60 percent) of total memory.
     - **TF**: Trace flags 111, 1118, 2371.

     You are strongly recommended to review existing configurations in this script and note their values in case a rollback is needed. However, do not edit this script.

- **codeinsight_db_creation_with_maintainenceplan.sql**—This script creates the database and schedules maintenance jobs. Specifically, it performs the following operations:
  - Creates a database with 4 data files and 1 log file.
  - Creates a new folder called `MSSQLDATA` on a non-OS disk. If only one drive exists, the database is created on the OS drive itself.
  - Creates a subfolder with the database name under the `MSSQLDATA` folder.
  - Creates a daily maintenance job to perform an Update Statistics every 6 hours (no downtime needed).
  - Creates maintenance job to perform an Update Statistics and Index Reorg every two weeks (no downtime needed). The default is to run at 10 pm per server time zone every two weeks.

  You can edit some settings in this script as described in Step 4.

3. Ensure that the SQL Server Agent is running.
4. Open the `codeinsight_serversettings.sql` script, and execute it.

   Do not edit this script.
5. Open the `codeinsight_db_creation_with_maintainenceplan.sql` script, edit the `@dbname` setting if necessary, and then execute the script.

   The default value for `@dbname` is **`fnciv7`**. To edit this setting, simply overwrite the current value with the preferred database name. If you provide a database name that already exists, the script execution will fail.
6. Create a user who has `READ` and `WRITE` permissions on the database (that is, the DBO role). This is the user who will access the Code Insight (SQL Server) database from the Code Insight application.

Network and Firewall Considerations

Configure the servers by specifying a fully qualified domain name (for example, *hostname.domain.com*) or IP address. Enable those port numbers used by FlexNet Code Insight in all of the firewalls. You may use the default port numbers listed below or configure the application to use custom ports.

Table 1-4 • Default Port Numbers Used by FlexNet Code Insight

Port #	Details
3306	MySQL database server access port
1433	SQL Server database server access port
8888/443	Tomcat (http/https)
465	External SMTP (mail) server
389	External authentication directory server (Active Directory/LDAP)
8005 and 8009	Tomcat Connector and Tomcat shutdown ports (local access only)

Setting the Open File Limit for Linux/Unix

The open file limit is a setting that controls the maximum number of open files for a specific user. The default open file limit is typically 1024, but can be set with the `ulimit` command by the root user. For FlexNet Code Insight to function properly in a Linux/Unix environment, the open file limit must be set to handle more than 50k files.

Important • *This procedure to increase open file size is absolutely essential for FlexNet Code Insight to function properly on Unix/Linux platforms.*

The following are some ways that open file limits are managed, depending on the user's role in the system:

- **soft limit**—Set in `/etc/security/limits.conf` by a normal user.
- **hard limit**—Set in `/etc/security/limits.conf` by root user.
- **system wide limit**—Set in `/etc/sysct1.conf` by root user.

Soft limits are the currently enforced limits, and hard limits are the maximum limits on the system. It is recommended that you log in as the root user so both types of limits may be set accordingly. Note that these limits must be defined as pertaining to a specific user or group, as described in the following procedure.

Task ***To set open file limits on a Linux RedHat system, do the following:***

1. In a terminal window, type **`ulimit -a`** to see a list of current file limits.
2. Locate the *open files (-n)* setting:
   - If the setting is less than 50K, continue to the next step.
   - If the setting is more than 50K, you do not need to perform this procedure.
3. Open the file `/etc/security/limits.conf`, and add the following entries for each specific user or group as needed:

   ```
   <userName> soft nofile 65536
   <userName> hard nofile 65536
   ```

   or

   ```
   @<groupName> soft nofile 65536
   @<groupName> hard nofile 65536
   ```

   Alternatively, you can substitute `<userName>` or `@<group name>` with the wildcard * for a default entry:

   ```
   * soft nofile 65536
   * hard nofile 65536
   ```

4. Save the file and log in again for the changes to take effect.
5. On the command line, type **`ulimit -a`**, and verify that the *open files (-n)* setting reads 65536.

Note • *Other distributions, such as a Ubuntu and CentOS may require a different setting. See instructions for your specific Linux distribution and shell type.*

Enabling Secure HTTP Over SSL

To implement SSL, a Secure Site SSL Certificate must exist for each Code Insight Core and Scan server that accepts secure connections. Refer to http://en.wikipedia.org/wiki/HTTP_Secure and http://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html for more details regarding HTTPS.

Use these instructions for enabling an HTTPS connection, including how to procure a certificate:

- Enabling an HTTPS Connection
- Purchasing a Secure Site SSL certificate
- Generating a Self-signed Certificate

Note • *For security, we recommend that FlexNet Code Insight always be installed over SSH.*

Enabling an HTTPS Connection

Use these instructions to enable the HTTPS connection.

Task ***To enable an HTTPS connection, do the following:***

1. Purchase a Secure Site SSL certificate, or generate your own self-signed certificate. The procedures for using a purchased certificate and for generating your own differ. Depending upon your type of certificate, consult one of the following sections:
   - Purchasing a Secure Site SSL certificate
   - Generating a Self-signed Certificate
2. Edit the `<CODEINSIGHT_ROOT_DIR>\tomcat\bin\catalina.bat` file (or the `catalina.sh` file depending on your operating system):

   `set -Dcodeinsight.ssl=true` (default value is `false`)
3. Back up the `<CODEINSIGHT_ROOT_DIR>\tomcat\conf\server.xml file` to another directory (outside of the `conf` directory), and then copy `server.xml` from `<CODEINSIGHT_ROOT_DIR>\tomcat\https` to `<CODEINSIGHT_ROOT_DIR>\tomcat\conf`.

   The `server.xml` file contains a default configuration that references a keystore at `<CODEINSIGHT_ROOT_DIR>\tomcat\codeinsight.jks`. You will need to update this information as needed for your certificate, as described in step 5.
4. In the `server.xml` file, locate the following text, and ensure that the `SSLEngine` value is **on**:

   ```
   <Listener
   className="org.apache.catalina.core.AprLifecycleListener"
   SSLEngine="on" />
   ```
5. In the `server.xml` file, locate for the following text that introduces the section describing the SSL certificate:

   `FNCI SSL: Edit this section to match your certificate information.`

This section shows the default values for the certificate:

```
<!-- FNCI SSL: Edit this section to match your certificate information -->
<Connector protocol="org.apache.coyote.http11.Http11Protocol"
   port="8888"
   minSpareThreads="25"
   enableLookups="false"
   disableUploadTimeout="true"
   acceptCount="100"
   maxThreads="150"
   maxHttpHeaderSize="8192"
   scheme="https"
   secure="true"
   SSLEnabled="true"
   keystoreFile="codeinsight.jks"
   keystorePass="codeinsight"
   keyAlias="codeinsight"
   keyPass="codeinsight"
   clientAuth="false"
   sslProtocol="TLS"
   ciphers="HIGH:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!kRSA"
/>
```

6. Update the following parameters in this section to reflect your certificate information:

```
keystoreFile: the file name of the keystore containing the certificate
keystorePass: the password of the keystore
keyAlias: the alias for the certificate entry in the keystore
keyPass: the password for the certificate entry
```

Note • *If the keystore and alias passwords are the same, you can specify keyPass, keystorePass or both.*

7. Restart the Tomcat server after making changes to the `server.xml` file or to a keystore. For more information, see Starting & Stopping Tomcat.

Purchasing a Secure Site SSL certificate

The following are two sources for purchasing a Secure Site SSL Certificate:

- http://www.verisign.com/ssl/buy-ssl-certificates/secure-site-ssl-certificates/index.html
- https://www.thawte.com/ssl-digital-certificates/ssl/index.html

Follow your vendor's instructions for generating a certificate signing request (CSR) and importing the certificate into the keystore.

Creating a Keystore for a Purchased Secure Site SSL Certificate--Example

The following is an example of a command to create a keystore for a Secure Site SSL Certificate on the server:

```
keytool -import -alias "<keyAlias>" -file <yourPurchasedCertificateFile> -keystore
<CODEINSIGHT_ROOT_DIR>\tomcat\<keystoreFile> -storepass "<keypass>"
```

Task ***To use a purchased Secure Site SSL Certificate, do the following:***

1. Export the certificate and import it into `cacerts`, which is in `<installDirectory>\jre\lib\security`.

   ```
   keytool -export -alias "<keyAlias>" -file <file>.crt -keystore <file>.jks
   keytool -delete -alias "<keyAlias>" -keystore cacerts
   keytool -import -alias "<keyAlias>" -keystore cacerts -file <file>.crt
   ```

   Note • *The default password for cacerts is changeit.*

2. (Optional) To check the contents of the keystore, enter the following command:

   `keytool -list -keystore cacerts shows keystore contents.`

3. Update the `<CODEINSIGHT_ROOT_DIR>\tomcat\conf\server.xml file` with values you provided in the command to create the keystore so Tomcat can access the generated certificate. See step 2 in the previous section, Enabling an HTTPS Connection.

Generating a Self-signed Certificate

Task ***To generate your own self-signed certificate with a keystore in place of a purchased one, do the following:***

1. Execute the following command found in the JDK:

   ```
   keytool -genkey -keyalg RSA -sigalg SHA256withRSA -alias "<keyAlias>" -keystore <keystoreFile> -
   storepass "<keypass>" -validity <numDays> -keysize 2048
   ```

2. Enter the server's host name or IP address when prompted, *What is your first and last name?*

3. Leave the rest of the prompts blank, except for the last one:

   `Is CN=<yourServerNameOrIPAddress>, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?`

   For this prompt, type **yes**.

4. Copy the generated keystore to `<CODEINSIGHT_ROOT_DIR>\tomcat\`.

5. Update the `<CODEINSIGHT_ROOT_DIR>\tomcat\conf\server.xml file` with values you provided in the command above so Tomcat can access the generated certificate. See step 2 in the previous section, Enabling an HTTPS Connection.

   If a self-signed certificate is used on the FlexNet CodeInsight server, each client machine that is used to access FlexNet Code Insight should add a certificate exception to the browser.

Using a Self-signed Certificate--Example

The following example uses a self-signed certificate and codeinsight for keystore, alias and passwords:

1. In `catalina.bat`, make the following changes:

   ```
   -Dcodeinsight.ssl=true
   tomcat\conf\server.xml replaced by the server.xml in tomcat\https
   ```

```
cd C:\mywork
keytool -genkey -keyalg RSA -sigalg SHA256withRSA -alias codeinsight -keystore codeinsight.jks -
storepass codeinsight -validity 3600 -keysize 2048
keytool -export -alias codeinsight -file codeinsight.crt -keystore codeinsight.jks
keytool -delete -alias codeinsight -keystore C:\FlexNetCodeInsight\jre\lib\security\cacerts
keytool -import -alias codeinsight -keystore C:\FlexNetCodeInsight\jre\lib\security\cacerts -file
C:\mywork\codeinsight.crt
keytool -v -list -keystore C:\FlexNetCodeInsight\jre\lib\security\cacerts -alias codeinsight
copy c:\mywork\codeinsight.jks C:\FlexNetCodeInsight\tomcat\
```

2. Restart Tomcat. For more information, see Starting & Stopping Tomcat.
3. Open a browser and enter **`https://<host>:8888/codeinsight`**.
4. Navigate to the System Configuration page, and update the scan server configuration.
   - Add a New scan server or select a scan server and edit it.
   - Set the Host name.
   - Set the Port to the https port.

Note • *You may need to accept browser warnings the first time that the application comes up; these messages should go away after the initial session is over.*

Configuring a Networking Proxy Server Connection

By default, FlexNet Code Insight uses automatic proxy server settings for any communications over the internet. However, FlexNet Code Insight can be manually configured to an enterprise networking proxy compliant with your company's IT policies.

Task ***To manually configure a proxy server connection, do the following:***

1. Navigate to the `tomcat/bin` folder. This folder resides in the directory where FlexNet CodeInsight is installed.
2. Open `catalina.bat` or `catalina.sh` for editing.
3. Locate the following line and uncomment it:

   ```
   rem set CATALINA_OPTS=%CATALINA_OPTS% -Dhttps.proxyHost=<HOST> -Dhttps.proxyPort=<PORT> :-
   Dhttps.proxyUser=<USER> -Dhttps.proxyPassword=<PASSWORD>
   ```

4. Set the following values for the proxy server:
   - **ProxyHost**—IP or Hostname of the proxy.
   - **PorxyPort**—Port being used for proxy.
   - **ProxyUser**—Username used to authenticate the proxy. Omit this value for a transparent proxy connection.
   - **ProxyPassword**—Password used to authenticate the proxy. Omit this value for a transparent proxy connection.
5. Restart the Tomcat server so the proxy server changes take effect. For information about restarting Tomcat, see Starting & Stopping Tomcat.

Installing FlexNet Code Insight

Use the following instructions to install FlexNet Code Insight:

- Gathering the Required Files
- Launching the FlexNet Code Insight Installer

Gathering the Required Files

While installing FlexNet Code Insight, you will be asked to provide your license key and directory paths for files that are needed for the application to function. In addition, you will choose the type of installation to perform. The following is a list of the items and information to collect before beginning the installation:

- The license key file, `codeinsight.key`. If you do not have a license key file, visit the Flexera Customer Community at https://flexeracommunity.force.com/customer/CCContactSupport.
- The appropriate JDBC driver connector file for the database:
  - **For MySQL:**
    - For MySQL 8.0, use `mysql-connector-java-8.0.x.jar`. You can download this file from https://dev.mysql.com/downloads/connector/j/.
    - For MySQL 5.7, use `mysql-connector-java-5.1.x-bin.jar`. You can download this file from http://dev.mysql.com/downloads/connector/j/5.1.html.
  - **For SQL Server**—The connector file `mssql-jdbc-6.4.0.jre8.jar`. If you do not have a connector file, download one from the Microsoft webpage: https://www.microsoft.com/en-us/download/details.aspx?id=56615.

  You must copy this connector file to your `tomcat/lib` folder.
- The type of installation you will perform:
  - **Standalone**—Configure your computer as both the core and scan server. This is the recommended configuration.
  - **Core**—Configure your computer as the core server.
  - **Scanner**—Configure your computer as the scan server.

  The Core Server controls your Web UI Client. The Scan Server is where actual scanning is performed.

Additionally, ensure that you have met the prerequisites listed in System Requirements.

Launching the FlexNet Code Insight Installer

After you create a database with remote access privileges, you can use the Installer to install FlexNet Code Insight in a Windows or Linux environment.

Note • *You can cancel the installation by clicking* ***Cancel*** *on any installation panel.*

Task ***To install FlexNet Code Insight, do the following:***

1. Download the FlexNet Code Insight installer from the Flexera Product and License Center:
   - For Windows, `FlexNetCodeInsight.exe`
   - For Linux, `FlexNetCodeInsight.bin`
2. Navigate to the directory where you downloaded the installer, and launch the installer.
3. Follow the prompts to install Code Insight.
4. When the installation is complete, do the following:
   a. Start the Tomcat server if it is not already running. See Starting & Stopping Tomcat.

      The recommended best practice is *not* to run Tomcat under elevated privileges.

   b. Launch Code Insight by following the procedures in Opening FlexNet Code Insight.

Important • *If the installation does not complete, contact* https://flexeracommunity.force.com/customer/CCContactSupport.

Running FlexNet Code Insight as a Service

Running FlexNet Code Insight as a service whenever your system starts up can save time. This section provides the procedure to configure FlexNet Code Insight as a service in both a Windows environment and a Linux (RedHat 7, CentOS 7) environment:

- In a Windows Environment
- In a Linux Environment

Recommended best practice is *not* to run Tomcat under elevated privileges.

In a Windows Environment

Perform the following procedure to run FlexNet Code Insight as a Windows service.

Task ***To run FlexNet Code Insight as a Windows service, do the following:***

1. Using the command prompt, navigate to:

   `<CODE_INSIGHT_ROOT_DIR>\tomcat\bin`
2. Stop the Tomcat server. See Starting & Stopping Tomcat.
3. Open the `service.bat` file with a text editor.

4. Set the `JRE_HOME` environment variable by adding this line at the beginning of the file. (You can copy this line from the `catalina.bat` file):

   ```
   set JRE_HOME=C:\<CODE_INSIGHT_ROOT_DIR>\jre
   ```

5. Under the `Set default Service name` comment, set the following parameters:

   - `SERVICE_NAME=CodeInsight`
   - `DISPLAYNAME=FlexNet Code Insight`

6. Change the `Description` to reflect the name of the service, which is *Code Insight*.

7. On the `JvmOptions` line, add the following to the list:

   - `-Dcodeinsight.ssl=false`
   - `-DcodeinsightInstallPath=<CODE_INSIGHT_ROOT_DIR>`

     The `<CODE_INSIGHT_ROOT_DIR>` is the directory path where FlexNet Code Insight is installed.

   Note • *Remember to separate the JvmOptions entries with a semi-colon (;).*

8. Change the `JvmMs` initial memory setting to `8192m`. The default entry is 128.

9. Change the `JvmMx` maximum memory setting to `16384m`. The default entry is 256.

10. Save the service.bat file and exit the text editor.

11. Execute the `service.bat install` command to install the Apache Tomcat Windows service.

12. When the service is installed, open **Windows Services** and search for the Service name you specified in step 4. In this case, it is *CodeInsight*.

13. Right click on the CodeInsight service and select **Start**.

In a Linux Environment

Perform the following procedure to run FlexNet Code Insight as a service on Linux (RedHat 7 or CentOS 7).

Task ***To run FlexNet Code Insight as a service in Linux, do the following:***

1. Create a file named `OpenSpecimen.service` with the following content.

```
[Unit]
Description=Tomcat Service OpenSpecimen.service
After=syslog.target network.target
[Service]
User=<userId>
WorkingDirectory=<fnci_install_path>
Type=forking
ExecStart=<fnci_install_path>tomcat/bin/startup.sh
ExecStop=/bin/kill -15 $MAINPID
[Install]
WantedBy=multi-user.target
```

Note the following:

- The `OpenSpecimen.service` file name is case-sensitive when referenced in the file content.
- Ensure that the Code Insight service is run under non-elevated privileges by specifying the `User` property (as shown). As an alternative, especially if the user ID starts with a number, you can specify the ID using the `login` argument in the `ExecStart` command, as in the example:

  ```
  ExecStart=/usr/bin/su --login <loginUserId> -c <fnci_install_path>tomcat/bin/startup.sh
  ```

2. Copy the `OpenSpecimen.service` file to the `/etc/systemd/system` directory.
3. Stop the Tomcat server. See Starting & Stopping Tomcat.
4. Execute the following command to notify systemd that the OpenSpecimen service has been added:

   ```
   $ sudo systemctl daemon-reload
   ```

5. Use the following commands to start, stop, or restart the OpenSpecimen service. (The `OpenSpecimen.service` file name is case-sensitive in the commands.)

   ```
   $ sudo systemctl start OpenSpecimen.service
   $ sudo systemctl stop OpenSpecimen.service
   $ sudo systemctl restart OpenSpecimen.service
   ```

6. Execute the following command to enable the starting of OpenSpecimen upon booting. (The `OpenSpecimen.service` file name is case-sensitive in this command.)

   ```
   systemctl enable OpenSpecimen.service
   ```

   From this point on, when you start your system, FlexNet Code Insight will start up automatically.

Starting & Stopping Tomcat

From time to time, it is necessary to start and stop the Tomcat server. For example, if this is the first time you have installed FlexNet Code Insight, or if you have recently upgraded FlexNet Code Insight or shut down your Tomcat server, you must restart it before you can connect to FlexNet Code Insight in a browser.

The recommended best practice is *not* to run Tomcat under elevated privileges.

Task ***To start the Tomcat server, do the following:***

1. Ensure the appropriate JDBC database connector file resides in `tomcat\lib`. See Gathering the Required Files.
2. Navigate to the directory where FlexNet Code Insight is installed and open the `tomcat\bin` directory (for example, `C:\FlexNetCodeInsight\tomcat\bin`).
3. Execute the `startup.bat file` for Windows or the `startup.sh` file for Linux. As the Tomcat startup runs, messages are displayed on the Tomcat console. The Tomcat startup may take several minutes to complete. When a startup message similar to the following appears in the Tomcat console, you can open FlexNet Code Insight in your browser:

   ```
   10-Aug-2017 10:06:34.796 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in
   58823 ms
   ```

Task ***To shut down the Tomcat server, do the following:***

1. Navigate to the directory where FlexNet Code Insight is installed and open the tomcat\bin directory. For example, `C:\FlexNetCodeInsight\tomcat\bin`.
2. Execute the shutdown.bat file for Windows or the shutdown.sh file for Linux.

Opening FlexNet Code Insight

FlexNet Code Insight runs in your web browser. This section explains how to start FlexNet Code Insight and access the **Dashboard**.

Task ***To open FlexNet Code Insight, do the following:***

1. Launch a web browser and navigate to the following URL, entering the server host name provided by your FlexNet Code Insight administrator:

   `http://<your_server_host_name>:PORTNUMBER/codeinsight/`

   For example, http://localhost:8888/codeinsight/.

   The FlexNet Code Insight Login page opens.

   Note • *If you are unsure about your server host name, contact your system administrator for guidance.*

2. Enter your Code Insight credentials in the **Username** and **Password**.

   Note • *The default login name is* **`admin`***; the default password is* **`Password123`***. However, your installation might require a different login name and password for the initial login. If you are unsure about what to enter, contact your system administrator for guidance.*

3. Click **Login**. The **FlexNet Code Insight Dashboard** appears.

   Important • *For increased security, it is highly recommended that you change the default password for* **`admin`** *on your first login. For details, Creating or Editing Users in the "Configuring FlexNet Code Insight" chapter.*

Roles and Permissions in FlexNet Code Insight

FlexNet Code Insight offers a set of user roles and permissions that enables your site to control access to Code Insight features and functionality. The initial Code Insight Administrator can assign system-level roles to users, including roles to manage Code Insight policies and to create and manage Code Insight projects. An Administrator can also assign the Administrator role to other users. Project Owners can assign project-specific roles to users to analyze and review project scan results and can also transfer project ownership to another user.

TheFlexNet Code Insight User Roles and Permissions appendix serves as a reference to the various Code Insight roles available and the permissions granted to each role. As you prepare use the Code Insight, refer to this appendix to determine the roles required to perform certain Code Insight functionality and the permissions the roles enable.

(Optional) Installing the Compliance Library

The FlexNet Code Insight Compliance Library (CL) is a library used by the codebase scan to perform exact-file and source-code fingerprint (snippet) matching. Code Insight compares elements of scanned codebase files with information contained in the CL to generate file-level evidence on which you can take action.

Using the CL is optional. The exact-file and source-code fingerprint (snippet) matching capabilities available with the CL are in addition to the Automated Analysis techniques basic to all scans to identify components, versions, licenses, and security vulnerabilities and to generate inventory.

Use the following instructions to install the CL. In general, install the CL on a drive accessible to the Code Insight Scan Server. For optimal performance, install the CL on the same machine as the Scan Server but on a different drive or volume from the one on which the Scan Server is installed. See Configuration Guidelines for more details.

Task ***To install FlexNet Code Insight, do the following:***

1. Download the Compliance Library (CL) installer from the Flexera Product and License Center:
   - For Windows, `FlexNetCodeInsightComplianceLibrary-`*`version`*`.exe`
   - For Linux, `FlexNetCodeInsightComplianceLibrary-`*`version`*`.bin`
2. Navigate to the directory where you downloaded the installer, and launch the installer.
3. Follow the prompts to install the CL.
4. When the installation is complete, navigate to the **Scan Servers** tab on the **Administration** page to configure the CL for use by future scans. Refer to Creating or Editing a Scan Server for instructions.

Uninstalling FlexNet Code Insight

An uninstaller for FlexNet Code Insight is available in the directory where the product is installed. The following procedures show you how to uninstall FlexNet Code Insight in a Windows and a Linux environment. Instructions are also provided to drop the SQL Server database used as the Code Insight database, should this action be necessary.

- Uninstalling on Windows
- Uninstalling on Linux
- Dropping the SQL Server Database

Uninstalling on Windows

Use the following procedure to uninstall Code Insight on a Windows machine.

Task ***To uninstall FlexNet Code Insight in Windows, do the following:***

1. Navigate to the directory where FlexNet Code Insight is installed.
2. Open the **Uninstall_FlexNetCodeInsight** folder.
3. Double-click **Uninstall FlexNetCodeInsight.exe**.
4. Follow the on-screen prompts to uninstall FlexNet Code Insight. The uninstall process will leave behind some files. Review them and delete as needed.

Uninstalling on Linux

Use the following procedure to uninstall Code Insight on a Linux machine.

Task ***To uninstall FlexNet Code Insight in* Linux*, do the following:***

1. Navigate to the directory where FlexNet Code Insight is installed.
2. Open the **Uninstall_FlexNetCodeInsight** folder.
3. Execute `Uninstall FlexNetCodeInsight` command and follow the on-screen prompts to uninstall FlexNet Code Insight. The uninstall process will leave behind some files. Review them and delete as needed.

Dropping the SQL Server Database

If you need to drop the SQL Server database used as the Code Insight database, follow this procedure. Dropping the database also drops its maintenance plans.

Task ***To drop the SQL Server database and its maintenance plans, do the following:***

1. If you have not already done so, download the `codeinsight_db_drop_with_maintainenceplan.sql` script. See Downloading the Scripts Needed to Set Up the SQL Server Database.
2. Open the script, and set the `@dbname` value to the name of the database to be dropped (if the value is not set to the correct name).
3. Execute the script.

Contacting Support

If you need further support, please submit your questions through our online **Customer Community** portal:

https://flexeracommunity.force.com/customer/

If you do not have a login to the Customer Community, you can request one on the **Login Request** page of our site:

https://flexeracommunity.force.com/customer/CCLoginRequest

If you are unable to use the steps above, please visit the following site for other options to reach out to Flexera Support:

https://flexeracommunity.force.com/customer/CCContactSupport

2

Configuring FlexNet Code Insight

After FlexNet Code Insight had been installed, the Administrator must perform a number of configuration tasks before the user can begin using Code Insight. This chapter describes these configuration tasks:

- Creating or Editing a Scan Server
- Managing Users
- Setting the Electronic Update Frequency
- Configuring an Email Server
- Configuring LDAP
- Configuring FlexNet Code Insight to Use Single Sign-On
- Managing Scan Profiles
- Setting Project Defaults
- About FlexNet Code Insight Server REST APIs

Note • *The first time you open FlexNet Code Insight, an electronic update will begin. The update can take 4 or more hours to complete. You cannot use the application to scan files until the update finishes. However, you can configure FlexNet Code Insight while the update is in progress.*

For information about the permissions granted to the Administrator role, see the FlexNet Code Insight User Roles and Permissions appendix.

Optionally, see About FlexNet Code Insight Server REST APIs in this chapter for information about Code Insight REST APIs that enable you to create your own administrative tool for managing scan operations and retrieving data from scan results.

Creating or Editing a Scan Server

The Scan Server scans the source code and binary files that make up your codebases to help you identify open source code that may expose your applications to compliance issues and security vulnerabilities. You must set up a scan server before scanning code.

Note • *FlexNet Code Insight supports the configuration of a single scan server only.*

For information about Code Insight scans in general, see "What Is a FlexNet Code Insight Scan?" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.

Task ***To create or edit your scan server, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Scan Servers** tab.
3. If you do not have a scan server configured, click **New**; or to edit your already-defined scan server, select it from the **Scan Servers** drop-down list, and click **Edit**. The **Scan Server** dialog appears.
4. Complete or update the fields on the dialog:
   - **Alias**—The name for the scan server.
   - **Host**—The IP address of the host computer for the scan server. If the scan server is on the same machine as the core server, enter *localhost*.
   - **Port**—The host port of the scan server. By default, the port is *8888*.
   - **CL Path**—(Optional) The path for the FlexNet Code Insight Compliance Library (CL), an optional Code Insight library downloaded from the Flexera Product and License Center (described in (Optional) Installing the Compliance Library). The CL is used by the scanner to perform exact-file and source-code fingerprint (snippet) matching. Code Insight compares elements of scanned codebase files with information contained in the CL to generate file-level evidence on which you can take action.

     You can leave this field blank and scan your code base using the scan profile labeled "Basic Scan Profile (without CL)". When you use this profile, only inventory from Code Insight's Automated Analysis feature is generated for the codebase. For more information, see the About Scanning without the Compliance Library.

     Keep in mind that, when you run a scan using the CL (that is, by specifying a valid CL path), you obtain a deeper, more comprehensive scan on your codebase.
   - **Codebase Path**—The path on the scan server where FlexNet Code Insight will store and manage all uploaded code. You should have adequate disk space to store the codebases. Recommended starting size for this directory is 500GB.

Note • *If you are unsure about what to enter in any of these fields, contact FlexNet Code Insight Support for guidance.*

About Scanning without the Compliance Library

By default, when FlexNet Code Insight scans a codebase, it uses the data in the Compliance Library (CL) to provide evidence of third-party code—exact-file matches and source-code fingerprint (snippet) matches—in your codebase.

However, if you do not have access to the CL (for example, you are running FlexNet Code Insight on a virtual machine or have not yet installed the CL) or do not want to enable your installed CL, leave the **CL Path** field blank on the **Scan Servers** tab on the **Administration** page (see Creating or Editing a Scan Server). You must then use the "Basic Scan Profile (without CL)" scan profile to perform a basic scan on your codebase. This scan uses Code Insight's Automated Analysis feature to perform the following:

- Generates inventory and detect vulnerabilities
- Finds evidence based on emails, URLs, and pre-defined search terms
- Employs all automated detection techniques

In the absence of a CL, FlexNet Code Insight will not detect exact-file matches and source-code fingerprint matches.

You can also create a custom basic scan profile with your own pre-defined search terms, as well as specify scan exclusions for folders or files to exclude from the codebase scan, such as **/.git or **/.hg.

For more information about the "Basic Scan Profile (without CL)" scan profile and about creating and managing scan profiles in general, see Managing Scan Profiles. For instructions on associating a scan profile with a project, see "Applying a Scan Profile to the Project" in the "Using FlexNet Code Insight" chapter in *FlexNet Code User Guide*.

Managing Users

The following topics describe how to manage FlexNet Code Insight users:

- Creating or Editing Users
- Managing User Permissions for System Activities
- Finding Users
- Disabling User Accounts

Creating or Editing Users

The following procedure describes how to create or edit users for your FlexNet Code Insight installation.

Note • *If you are using an LDAP server to synchronize the user data, you can skip this procedure. To configure an LDAP server, see Configuring LDAP.*

Task ***To create or edit a user, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Users/Permissions** tab, which lists all current users.

3. To create a new user, click **Add User**; or to edit an existing user, click the Edit icon  .

   The **Add User** or **Edit User** dialog appears.

4. Enter information in the fields to create or edit the user:

   - **Login**—The user's login name.
   - **First Name**—The user's first name.
   - **Last Name**—The user's last name.
   - **Email**—The user's email address.
   - **Password**—The user's password, which should be a minimum of 8 characters with no spaces and have at least one number and one capital letter.
   - **Password Confirm**—Reenter the password from the field above.
   - **Question**—A security question that can be answered by the user to retrieve a lost password. The question must be a minimum of 3 characters.
   - **Answer**—The answer to the security question.

5. When you finish entering information for the user, click **Submit**. The **Success** dialog appears, telling you that the user has been saved.

6. Click **OK**. If you created a user, the user will appear in the list.

7. To assign permissions to administrate Code Insight, manage policies, and create projects, see the next section, Managing User Permissions for System Activities.

Managing User Permissions for System Activities

Use the procedures described in this section to grant or revoke the following types to user permissions used to manage system-wide activities:

- **Administrators**—Grants the user permission to create and manage users and configure FlexNet Code Insight at the global level.
- **Manage Policy**—Grants the user permission to manage policies that automate the inventory review process—that is, automatically mark published inventory items as approved, rejected, or requiring a manual review—without the need for a manual review.

  Policy details are described in "Managing Policies" in the "Using FlexNet Code Insight" chapter in *FlexNet Code User Guide*.

- **Create Project**—(Displayed only if you selected **No** for **Allow all users to create projects?**) Grants the user permission to create projects and project folders. (Users become the project owner of each project they create.)

  A **Create New** button, enabling users to create projects and project folders, is visible on the **Projects** page for only those users granted this permission. Project and folder creation is described in "Creating a Project" and "Managing Items in the Project List" in the "Using FlexNet Code Insight" chapter in *FlexNet Code User Guide*.

Note • *In addition to these permissions, roles can be assigned to users at the individual project level, as described in the FlexNet Code Insight User Guide.*

See the topics in this section for more information:

- Grant System Permissions to Users
- Revoke User Permissions

Grant System Permissions to Users

Follow these steps to grant one or more permissions to individual users.

Task ***To grant permissions to users, do the following:***

1. Navigate to the **Users/Permissions** tab on the **Administration** page. (For instructions on getting to this tab, see the initial steps in Creating or Editing Users.)
2. Click **Manage Permissions** to open the **Manage Permissions** dialog.
3. Select the **Yes** or **No** option for **Allow all users to create projects?** to determine whether all or selected users will have permission to create projects. If you select **No**, a **Create Project** pane is added to the dialog to enable you to select the users to which to grant this permission. (The default is **Yes**, allowing any user to create projects.)
4. To grant a permission to a given user, drag and drop the user name from the **Select Users** list to the desired permission pane (**Administrators**, **Manage Policy**, or **Create Project**).
5. Repeat this step as necessary to assign multiple permissions to the same user or to grant permissions to another user.
6. Click **Close** to return to the **Users/Permissions** tab.

Revoke User Permissions

Follow these steps to revoke a user's permissions.

Task ***To revoke user permissions, do the following:***

1. Navigate to the **Users/Permissions** tab on the **Administration** page. (For instructions on getting to this tab, see the initial steps in Creating or Editing Users.)
2. Click **Manage Permissions** to open the **Manage Permissions** dialog.
3. To remove a permission from a user, navigate to the desired pane, and click ✖ next to the user name to remove the user from the pane.
4. Once you have revoked the necessary permissions, click **Close** to return to the **Users/Permissions** tab.

Finding Users

As a system administrator or project owner, you might need to find FlexNet Code Insight users to manage their permissions. You can search for users on the **Users** tab or on the **Summary** tab for the project.

Task ***To find users, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Users** tab.
3. In the **Enter Search Criteria** field, enter a character string by which to search user information in any of the fields.
4. Click **Search**.

Disabling User Accounts

FlexNet Code Insight supports disabling user accounts in the browser.

Note • *The Admin user account is created automatically; it cannot be disabled.*

Task ***To disable user accounts, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Users** tab.
3. Click the **Edit** icon ( ) in the **Actions** column for the user account you want to disable. The **Edit User** dialog appears.
4. Select the **Disable Account** checkbox, and click **Submit**. The **Success** dialog appears.
5. Click **OK**. The user account is now disabled. The user will receive the message, "`Invalid Username and/or Password. If you believe you entered a valid user, please contact your System Administrator`" when attempting to log into FlexNet Code Insight."

Setting the Electronic Update Frequency

Frequent Electronic Updates enable you to receive the latest vulnerability or other component information for your product as quickly as it is available. By default, an incremental Electronic Update is set to run automatically on a daily basis, but FlexNet Code Insight enables you to edit the frequency at which the automatic update runs, as required for your site.

Additionally, you have the option to force an incremental or full Electronic Update whenever needed.

Refer to the following for more information:

- Resetting the Frequency of the Regularly Scheduled Update
- Forcing an Electronic Update

Resetting the Frequency of the Regularly Scheduled Update

An incremental Electronic Update is set, by default, to run automatically at 1 am each day, but you can change the frequency at which this update runs.

Note • *Codebase scans cannot be performed during the Electronic Update process, but a scan that is already underway will not be interrupted when an automatic update process is scheduled to begin. The update will be queued and automatically run based on queue order.*

Task ***To reset the Electronic Update frequency, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Electronic Updates** tab.
3. From the first dropdown in the **Update Frequency** section, select the frequency at which to run the Electronic Update:
   - **Never**—If you select **Never**, no automatic Electronic Update is run regularly. (Selection of this option hides the other **Update Frequency** dropdowns.)

     If you need to run an update, you can manually trigger an incremental or full update. See Forcing an Electronic Update for details.
   - **Daily**—If you select **Daily**, select a clock time from the second, or "time", dropdown down to indicate at what time of day to run the update.
   - **Weekly**—If you select **Weekly**, select a clock time from the "time" dropdown and a weekday from the **Select a day...** dropdown to indicate when to run the weekly update.
4. When you have finished setting the update frequency, select **Save**. A prompt appears to notify you that your edits have been saved.

Forcing an Electronic Update

You can manually trigger an incremental or full Electronic Update any time between the regularly scheduled automatic updates. You might need to so, for example, if the most recent update did not complete properly or if you cannot wait for the next scheduled update to determine critical information, such as whether any new vulnerabilities are impacting your product.

Additionally, if you have selected *not* to run automatic Electronic Updates (as described in Resetting the Frequency of the Regularly Scheduled Update), you can use this procedure to force an update as needed.

Note • *Forcing an Electronic Update might cause an unexpected delay in starting a codebase scan. However, if a scan is already underway when the forced update process is triggered, the update is queued and automatically run based on queue order.*

Task ***To force an Electronic Update, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Electronic Updates** tab.
3. Select either process:
   - **Request an incremental update**—Click **Schedule Update** to request an incremental update to obtain any changes since the last update. If Code Insight determines that changes have occurred since the last update, the update is run immediately. If no changes exist, no update is run.
   - **Force a full update**—Select the **Force Full Electronic Update** option and then click **Schedule Update** to force a full update whether or not changes have occurred since the last update. Use this option with caution as the full update requires more overhead than an incremental update.

Configuring an Email Server

FlexNet Code Insight can send email alerts that are triggered by certain events. For example, when a scan completes or when a new vulnerability is detected in the project inventory. It is highly recommended that the email server configuration be set up for the application. Email server configuration is available in FlexNet Code Insight in the Administration tabs. This section provides the procedure for configuring email.

Task ***To configure your email server, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Email Server** tab.
3. Enter information and make selections in the fields:
   - **Enable Email Server**—Select **Yes** to enable FlexNet Code Insight to use the email server or **No** to leave it disabled. The default is **No**. The rest of the fields on this page are not available until you select **Yes**.
   - **Sender's Email Address**— Enter the email address of the sender.
   - **SMTP Host Name**—Enter the SMTP host name.
   - **SMTP Host Port**—Enter the port number of the SMTP host.
   - **SMTP User Name**—Enter the SMTP user name. This field can be left blank for anonymous SMTP configuration.
   - **SMTP User Password**—Enter the SMTP user password. This field can be left blank for anonymous SMTP configuration.

- **Enable SMTP over TLS**—Select **Yes** to use Transport Layer Security (TLS) to secure email over SMTP or select **No** to leave this option disabled.

4. Click **Save** to save your settings.

Configuring LDAP

The LDAP option allows you to use an LDAP server to import user name data into FlexNet Code Insight and for authentication, as described in the following topics:

- Synchronizing User Name Data
- Setting Up a User Search Filter
- Sample Search Query
- Sub-tree Search
- Server Paging
- User Authentication
- LDAP over SSL
- Implementing LDAP

Synchronizing User Name Data

FlexNet Code Insight provides the ability to import user name data from LDAP. This section explains the type of user name data that is imported.

User Metadata

The metadata for each user (name, email, etc.) is pulled from LDAP and refreshed in the FlexNet Code Insight database at a regular frequency via a scheduler module running within FlexNet Code Insight. The data synchronization is a one-way pull from LDAP into the FlexNet Code Insight database. This action overwrites the existing data in the FlexNet Code Insight database. User data for those users that do not exist in LDAP is not affected by this process.

Disabled Users

Users who are disabled in FlexNet Code Insight will still have their data synchronized with LDAP, but will have the disabled flag set to “true” and will not be granted access to the application.

Setting Up a User Search Filter

To pull only the required users into FlexNet Code Insight, it is important to configure the **LDAP Search Base** and **LDAP Search Query** entries, which appear on the **LDAP** tab of the FlexNet Code Insight user interface, properly. The **LDAP Search Base** is typically the root node under which you can store all the desired users. The **LDAP Search Query** allows LDAP queries based on user attributes. Best practice is to create a FlexNet Code Insight system-specific group and make all of the desired users part of this group.

Sample Search Query

LDAP search queries can be entered in the **LDAP Search Query** field on the **LDAP** tab. For example, the following query pulls only desired users into FlexNet Code Insight:

```
(&(objectClass=person)(memberOf=CN=Code InsightGroup,CN=Users,DC=ad,DC=Code Insight,DC=com))
```

Sub-tree Search

The **Search Sub-tree** option on the **LDAP** tab controls whether to enable deep searches through the subtree of the path defined by **LDAP Base** + **LDAP Search Base**. While helpful in locating users in certain cases, a deep search can negatively affect performance (and therefore, by default, is not enabled).

Server Paging

LDAP and Active Directory support server paging controls the number of records the system is pulling at any given time. Configure the **LDAP Page Size** entries as desired. The default page size is 1000.

Note • *SunOne Directory Server does not support server paging in certain releases http://kb.globalscape.com/KnowledgebaseArticle10218.aspx. If you are using SunOne Directory Server, ensure that server paging is disabled.*

User Authentication

You can use an existing LDAP server to verify users when they log into FlexNet Code Insight. FlexNet Code Insight does not store LDAP passwords. All authentication happens on the LDAP server. After an LDAP user enters a username and password, the credentials are sent to the LDAP instance. If LDAP confirms that the user is valid, FlexNet Code Insight grants access.

Note • *If you configure LDAP to provide login security, the built-in FlexNet Code Insight login security will not be used.*

LDAP over SSL

SSL provides data encryption security for user information passed over the network. You must use ldaps://URL with 636 port, which is the default dedicated port for SSL.

Implementing LDAP

This section explains the basic procedure for implementing LDAP in FlexNet Code Insight. For detailed descriptions of the fields on the LDAP tab, see the “LDAP tab” topic in the online help or *FlexNet Code Insight User Guide*.

Task ***To implement LDAP, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **LDAP** tab.
3. Select **Yes** in the **Enable LDAP** field and complete the rest of the fields on the **LDAP** tab. See "LDAP Tab" in the online help or in the *FlexNet Code Insight User Guide* for descriptions of all the fields.
4. (Optional) Select **Test LDAP Server Connection** to ensure that FlexNet Code Insight is properly connected to the LDAP server. The connection will be tested with the values displayed in the fields on the **LDAP** tab.
5. When you finish entering information in the fields, select **Save** to save your changes to the LDAP configuration.
6. (Optional) Select **Sync Now** to save your settings and synchronize them with the user data on the LDAP server. If you do not select **Sync Now**, the user synchronization will be done at the time specified in the **LDAP User Sync Frequency** field.

Configuring FlexNet Code Insight to Use Single Sign-On

Single sign-on (SSO) is an authentication service that enables a user to use one set of credentials (usually a name and password) to access multiple applications. This service involves an exchange of SAML (Security Assertion Markup Language) protocol messages between the user, the identity provider, and the service provider.

The Identity Provider (also called an IdP) is any SSO service, such as Okta, Ping Federate, and others, offering SAML authentication services. The Service Provider (also called an SP) is an application, such as FlexNet Code Insight, that is configured to participate in the SSO service. When a Service Provider user logs in using credentials for an SSO session, a SAML message is sent to the Identity Provider, requesting user authentication. If the user password is valid, the Identity Provider returns a SAML message, stating that the user is logged in at the Identity Provider. The user, in turn, is logged into the Service Provider.

The FlexNet Code Insight administrator can use the instructions in these sections to configure Code Insight as a Service Provider in an SSO session:

- Prerequisite Tasks for Configuring Code Insight for SSO
- Configuring Code Insight for SSO
- Log In Using SSO Credentials

Prerequisite Tasks for Configuring Code Insight for SSO

Perform the following tasks before configuring Code Insight for SSO:

- Configure HTTPS on the FlexNet Code Insight Server
- Set Up SSO Users

Configure HTTPS on the FlexNet Code Insight Server

The HTTPS communication protocol must be used to exchange SAML messages between the SP and IdP. For instructions on configuring HTTPS on the Code Insight server, see Enabling Secure HTTP Over SSL in the "Installing FlexNet Code Insight" chapter.

The keystore that you use to configure HTTPS can be used for SSO configuration. Alternatively, you can create a separate keystore for SSO, using the same instructions found in Enabling Secure HTTP Over SSL.

Set Up SSO Users

You can define SSO users for Code Insight with or without LDAP.

With LDAP

If you intend for SSO to integrate with your LDAP server for user access to Code Insight, follow these rules:

- Make sure that Code Insight and the Service Provider are configured for the LDAP server. For instructions to configure Code Insight, see Configuring LDAP.

  To configure the Service Provider, follow the Service Provider instructions.
- When setting up users on the LDAP server, ensure that the user's login is the user's email address.
- Synchronize users from the LDAP server to the Identity Provider first, using the Identity Provider's instructions. Then synchronize the users from the LDAP server to Code Insight. See Configuring LDAP.

Without LDAP

If you do not use LDAP, you must manually create the SSO users both in FlexNet Code Insight (see Managing Users) and at the Identity Provider site, ensuring that the user information is the same in both locations.

Ensure that the user's login is the user's email address.

Configuring Code Insight for SSO

Follow these steps for configuring Code Insight for SSO:

- Step 1: Copy the Directory That Will Contain Provider Metadata
- Step 2: Prepare the Environment Properties File
- Step 3: Configure the SSO Common Properties File
- Step 4: Customize the Sample Service Provider Metadata File
- Step 5: Obtain the Identity Provider Metadata File

Note that, in these instructions, *SCA_install_home* refers to the Code Insight installation location.

Step 1: Copy the Directory That Will Contain Provider Metadata

Copy the `security` directory from *SCA_install_home*`/samples/sso/config/core` to *SCA_install_home*`/config/core`.

This directory will serve as the storage location for the Service Provider and Identity Provider metadata files, as described in Step 4: Customize the Sample Service Provider Metadata File and Step 5: Obtain the Identity Provider Metadata File.

Step 2: Prepare the Environment Properties File

This step prepares the `env.properties` file to enable SSO on the Code Insight server.

Task ***To prepare the "env.properties" file, do the following:***

1. Copy the `env.properties` file from *SCA_install_home*`/samples/sso/config` to *SCA_install_home*`/config/core`.
2. In a text editor, open the *SCA_install_home*`/config/core/env.properties` file, and ensure that the value of the following property to **`sso`**.

   ```
   spring.profiles.active=sso
   ```

3. Save the file.

Step 3: Configure the SSO Common Properties File

This step configures the `core.sso.common.properties` file to enable SSO on the Code Insight server.

Task ***To prepare the "core.sso.common.properties" file, do the following:***

1. Copy the `core.sso.common.properties` file from *SCA_install_home*`/samples/sso/config` to *SCA_install_home*`/config/core`.
2. In a text editor, open the *SCA_install_home*`/config/core/core.sso.common.properties` file. The following shows the file contents:

   ```
   ## this file contains all sso placeholder values.
   saml.keystore=file:///c:/<path>/keystore.jks
   saml.keystore.password=keysore_password
   saml.keystore.alias=keystore_alias
   saml.keystore.alias.password=keystore_alias_password

   # for extendedMetadata configuration
   saml.metadata.local=true
   saml.metadata.alias=
   saml.metadata.idpDiscoveryEnabled=false
   saml.metadata.idpDiscoveryURL=
   saml.metadata.idpDiscoveryResponseURL=
   saml.metadata.ecpEnabled=false
   saml.metadata.securityProfile=metaiop
   saml.metadata.sslSecurityProfile=pkix
   saml.metadata.sslHostnameVerification=default
   saml.metadata.signingKey=keystore_alias
   ```

```
saml.metadata.signingAlgorithm=null
saml.metadata.signMetadata=false
saml.metadata.encryptionKey=keystore_alias
saml.metadata.tlsKey=
#private Set<String> trustedKeys=
saml.metadata.requireLogoutRequestSigned=false
saml.metadata.requireLogoutResponseSigned=false
saml.metadata.requireArtifactResolveSigned=false
saml.metadata.supportUnsolicitedResponse=true
#for SP
saml.entity.id=ww:xx:yy:zz
saml.base.url=https://myhost.mycompany.com:8443
```

3. Update the properties (highlighted above) required for Service Provider security and identification, and then save the file. The properties that you need to edit or that require explicit configuration are described in this table:

SSO Property	Description
saml.keystore	Enter the path and name of the keystore that you created for SSO. This can be the same keystore that you are using for HTTPS or a different one. See Configure HTTPS on the FlexNet Code Insight Server in the "Installing FlexNet Code Insight" chapter for more information.
saml.keystore.password	Enter the password for the keystore.
saml.keystore.alias	Enter the alias defined for the private key contained in the keystore.
saml.keystore.alias.password	Enter the password for the private key alias.
saml.metadata.alias	Provide your metadata alias, if one exists; or leave this field blank (or enter `defaultAlias`) to use the default metadata alias.
saml.metadata.idpDiscovery URL	Leave this field blank. Do not enter `null`.
saml.metadata.idpDiscovery ResponseURL	Leave this field blank. Do not enter `null`.
saml.metadata.signingKey	Enter the path and name of the keystore you created for SSO. (This is the same value entered for the `saml.keystore` property.)
saml.metadata.encryptionKey	Enter the path and name of the keystore you created for SSO. (This is the same value entered for the `saml.keystore` property.)
saml.metadata.tlsKey	Enter the alias of private key generated for SSL/TLS client authentication, if one exists; or leave this field blank to use the default TLS key alias.
saml.entity.id	Enter a unique identifier for your Code Insight server as a Service Provider. The recommended value is the hostname for the Code Insight server. Note that, even though the server's hostname is the recommended value, the entity ID is an immutable value identifying the Service Provider in an SSO session; it is not used to identify a location.

SSO Property	Description
saml.base.url	The HTTPS URL handling the Service Provider's user sign-in requests. This is usually the URL for your Code Insight server in `HTTPS://myhost.mycompany.com:port` format. Note that the default port for the Code Insight server is 8443.

Step 4: Customize the Sample Service Provider Metadata File

This step customizes the sample Service Provider metadata file for your Code Insight server.

Task ***To customize the sample Service Provider metadata file, do the following:***

1. In a text editor, open the *SCA_install_home*`/config/core/security/SPMetadata.xml` file.
2. Update the following properties, and save the file:

SSO Property	Description
entityID="*ENTITY_VALUE*"	Replace ENTITY_VALUE with the same entity ID as the one you provided the `env.properties` file in Step 2: Prepare the Environment Properties File.
SingleLogoutService... *FULLY_QUALIFIEDHOSTNAME*...	Replace *FULLY_QUALIFIEDHOSTNAME* with the fully qualified hostname of the Code Insight server.
AssertionConsumerService... *FULLY_QUALIFIEDHOSTNAME*...	Replace *FULLY_QUALIFIEDHOSTNAME* with the fully qualified hostname of the Code Insight server.

Step 5: Obtain the Identity Provider Metadata File

This final step in setting up SSO for Code Insight is to obtain the Identity Provider metadata file. The Identity Provider might require that you send the Code Insight `SPMetadata.xml` file (set up in Step 4: Customize the Sample Service Provider Metadata File) in order to provide the Identity Provider metadata file.

Alternatively, you might be required to generate the Identity Provider metadata file using the Identity Provider UI. You will need to provide the single-sign-on URL for Code Insight (also specified in the `SPMetadata.xml`):

`https://myhost.mycompany.com:8443/codeinsight/saml/SSO`

Task ***To obtain the Identity Provider metadata, do the following:***

1. Follow the Identity Provider's instructions for obtaining the Identity Provider metadata.
2. Once you obtain the Identity Provider metadata, save it as `IDPMetadata.xml` in the *SCA_install_home*`/config/core/security` directory.

Log In Using SSO Credentials

Once you complete the steps described in this section, Code Insight users defined as SSO users should be able to log in to an SSO session managed by the Identity Provider and obtain access to Code Insight.

Managing Scan Profiles

The following topics describe how to manage scan profiles:

- Creating or Editing Scan Profiles
- Scan Profile Fields
- Creating Exclusion Patterns for Scan Profiles

Creating or Editing Scan Profiles

A scan profile is a set of predefined scan settings that are grouped together and then applied at scan time. (To enable this application, the Project Owner associates the scan profile with a project, as described in "Applying a Scan Profile to the Project" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code User Guid*e.)

By default, FlexNet Code Insight provides the following scan profiles. (See Scan Profile Fields section for a summary of the scan functions included in each of these profiles.)

- Basic Scan Profile (without CL)
- Standard Scan Profile
- Comprehensive Scan Profile

In most cases, the pre-defined scan profiles are enough to get started. However, if they do not meet your needs, you can create your own custom scan profiles. When a scan profile is created, the data from the Standard Scan Profile is copied, including any search terms and exclusions. However, you can update any of this information in the scan profile you are creating.

You can also edit information in existing scan profiles (except the Standard Scan Profile). Note the following:

- Scan profiles changes can result in costly rescans, especially when settings involved with source-code matching change. For details, refer to the "Rescanning Your Codebase" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code User Guid*e.
- Scan profiles changes do not affect the current scan. Changes are applied to the next scheduled scan.

The following procedure describes how to create or edit a scan profile.

Task ***To create or edit a new scan profile, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Scan Profiles** tab.

3. Click **New** or **Edit** next to the drop-down field listing the existing scan profiles. The **Create** (or **Edit**) **Scan Profile** dialog appears.
4. Complete the fields on the dialog. See the next section, Scan Profile Fields.
5. Click **Save** to save the scan profile.

Scan Profile Fields

The following table summarizes the function of each field in the scan profile. It also notes which fields are valid for the default scan profiles shipped with Code Insight:

Table 2-1 • Scan Field Descriptions and Default Scan Profile Support

Field	Description	Basic	Standard	Comprehensive
Perform Package/License Discovery in Archives	Select this option to have the scanner recursively perform package discovery and license detection within all archive files encountered in the project codebase. By default, this option is selected.	X	X	X
Dependency Support	Determine the level of dependency scanning to be performed by the scanner. The available options include: • **No Dependencies**: Only top-level inventory items are reported. (Default) • **Only First Level Dependencies**: Only first-level (or direct) dependencies are reported along with top-level inventory items. • **All Transitive Dependencies**: All first-level and transitive dependencies are reported along with top-level inventory items. The scanner calls out to the relevant package management repository to obtain transitive dependency information. For a description of Code Insight dependency support for supported ecosystems, see the "Automated Analysis" chapter in the *FlexNet Code Insight User Guide*.	X	X	X
Automatically Add Related Files to Inventory	Select this option to have the system associate additional files to existing inventory items based on the data available in automatic detection rules.	X	X	X
Exact Matches	Select this option to have the scanner record exact matches for scanned files based on data from the Compliance Library (CL).		X	X

Table 2-1 • Scan Field Descriptions and Default Scan Profile Support (cont.)

Field	Description	Basic	Standard	Comprehensive
Source Code Matches	Select this option to have the scanner record source code matches for scanned files based on data from the Compliance Library (CL).			X
Include System Identified Files	(Available only when **Source Code Matches** is selected) Select this option if you want the scanner to perform source-code matching for files that have already been associated with one or more inventory items through automated analysis.			X
Include Files with Exact Matches	(Available only when **Source Code Matches** is selected) Select this option if you want the scanner to perform source-code matching for files that have already been identified as having exact matches.			X
Search Terms	Provide a list of search terms to be used in the scan.	X	X	X
Scan Exclusions	Provide a list of file extensions to be excluded from the scan. Also see Creating Exclusion Patterns for Scan Profiles.	X	X	X

Note *• Comprehensive and Standard Scan Profiles rely on data stored in the Compliance Library (CL) to detect evidence for Exact Matches and Source Code Matches.*

Creating Exclusion Patterns for Scan Profiles

Flex Net Code Insight provides the ability to create exclusion patterns for use in your scans and to add them to your scan profile in **Create** (or **Edit**) **Scan Profile** page. This section provides information about the syntax required when creating exclusion patters and examples of valid exclusion patterns.

Flex Net Code Insight uses Apache Ant path-style syntax to exclude files during scanning. Patterns are paths that are relative to a base directory. Only files found in or below the base directory are considered for exclusion. For in-depth information about *ant* exclusion patterns, see `https://ant.apache.org/manual/dirtasks.html`.

Note *• Exclusion patterns are not validated.*

Using the Single Asterisk (*) and Question Mark (?)

Using a single asterisk (*) matches zero or more characters. Using the question mark (?) matches one character. If you create an exclusion pattern of *.xml, and add it to the list of Scan Exclusions in FNCI, your scan will exclude files such as `x.xml, FooBar.xml`, `codeinsight.xml` but not `codeinsight.jar` because it does not end with `.xml`.

If you create an exclusion pattern of `?.codeinsight` and add it to your list of Scan Exclusions in FNCI, your scan will exclude files such as `x.codeinsight` and `A.codeinsight`, but not `xx.codeinsight` or `aaa.codeinsight` because neither has just one character before `.codeinsight`. In other words, `xx.codeinsight` and `aaa.codeinsight` *will* appear in scan results if they are in your codebase.

Note • *You can combine asterisks (`*`) and question marks (`?`) in your exclusion patterns.*

Using Double Asterisks

Double asterisks (**) span multiple directory paths If you create an exclusion pattern of **/codeinsight, the files in the aa/bb/cc/codeinsight directory structure will be excluded from the scan.

Sample Exclusion Patterns

The following are some sample patterns that can be used with FNCI:

Table 2-2 • Sample Exclusion Patterns and Descriptions

Pattern	Description
****/SVN/***	Excludes all the files in the `SVN` directories that are located anywhere in the directory tree (e.g., `SVN/Repository`, and `apache/SVN/Entries`) from the scan. But `org/apache/SVN/foo/bar/Entries` will be included in the scan.
/ePortal-2.0/src/**	Excludes all the files in the `/ePortal-2.0/src/**` directory tree (e.g., `/ePortal-2.0/src/index.html`, and `/ePortal-2.0/src/test.xml`). But `/ePortal-2.0/src/**xyz.java` will be included in the scan.
****/git**	Exclude all files in aa/bb/cc/git.

Note • *Exclusion patterns are not validated by FNCI. Please test your pattern externally.*

Note • *If a pattern ends with `/` or `\`, double asterisks (`**`) are appended. For example, `codeinsight/data/` is interpreted as `codeinsight/data/**`.*

Setting Project Defaults

The settings on **Project Defaults** tab on the **Administration** page work in conjunction with a project's policies to configure the automation of review, remediation, and status notification processes for published inventory. These settings, which are global across all projects but can be overridden at the project level, are used to set up the following:

- Automatic creation of manual review tasks for inventory items not reviewed by policy during the publication performed as part of a scan. The tasks are automatically assigned to a default legal or security contact (defined at the project level, as described in "Updating Inventory Review and Remediation Settings for a Project" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*).
- Automatic creation of remediation tasks and associated external work items for inventory that is rejected either automatically by policy or during manual publication by an analyst. The tasks are automatically assigned to a default development contact (defined at the project level, as described in "Updating Inventory Review and Remediation Settings for a Project" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*).
- Automatic rejection of published inventory impacted by new vulnerabilities detected in the latest scan or Electronic Update.
- The automatic generation of email notifications only (instead of assigned tasks), which are sent to the project owner as alerts concerning the rejected or non-reviewed published inventory items.

For more information about policies, refer to "Managing Policy Profiles" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.

Task ***To set project defaults, do the following:***

1. On the **FlexNet Code Insight Dashboard**, click **administration**. The **Administration** page appears with a list of side tabs.
2. Select the **Project Defaults** tab.
3. Update global default values as needed, using the following table for field descriptions.

Section/Field	Description
Automated Review Options	

Section/Field	Description
automatically reject inventory items impacted by a new vulnerability that violates your policy	Determine what action the system should take for published inventory affected by a new security vulnerability discovered during a post-publication scan or Electronic Update. The selected action applies to both non-reviewed and previously approved inventory items on the **Project Inventory** tab. • Select this checkbox to automatically reject those project inventory items impacted by a new security vulnerability only if this vulnerability has a CVSS score or severity *greater than* the thresholds configured as policy for the Code Insight project. For each inventory item rejected due to a new security vulnerability, the  icon and a tip are added to indicate the status change and its reason. If a new vulnerability does not exceed policy thresholds, the current status of the inventory item is not affected. • Leave the checkbox unselected to retain the current status of inventory items impacted by the new vulnerability. For information about policies, refer to "Managing Policy Profiles" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.
Manual Review Options	
What should happen if inventory items are not reviewed by policy?	Determine what action should be triggered for those inventory items that are *not* affected by policy (and therefore have a **Not Reviewed** status) during the publication of inventory either as part of a scan or manually by a user: • **do nothing**—Simply show the status of the inventory item as **Not Reviewed** on the **Project Inventory** tab. • **send an email notification to the project owner**—Automatically send an email to the project owner, stating the need for a manual review of the item. The value for **Select the minimum priority...** (described in the next table entry) affects this option. • **automatically create a manual review task**—Automatically create a manual review task assigned to the default legal or security reviewer, and send an email, notifying the reviewer about assigned task. (Default reviewers are defined at the project level on the **Edit Project** dialog, as described in "Updating Inventory Review and Remediation Settings for a Project" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.) Information about managing such a task to track the progress of a manual review is found in "Creating and Managing Tasks for Project Inventory" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.) The value for **Select the minimum priority...** (described in the next table entry) affects this option.

Section/Field	Description
Select the minimum priority to perform the action selected above	(Enabled when an option other than **do nothing** is selected for the previous field.) Select the minimum inventory priority (**P1**, **P2**, **P3**, or **P4**) to which the value for the previous field applies. For example, if the previous field is set to **send an email notification to the project owner** and minimum priority is set to **P3**, then the email notification will be sent for only those non-reviewed inventory items with a P1, P2, or P3 priority. No email notification will be sent for P4 inventory items.  ***Note*** • *This option has no effect on the* ***do nothing*** *value.*

What should happen if inventory items are rejected?	Determine what action should be triggered for those inventory items that are automatically rejected by policy during the publication of inventory either as part of a scan or manually by a user:
	
	• **do nothing**—Simply show the status of the inventory item as Reject on the Project Inventory tab.
	
	• **send an email notification to the project owner**—Automatically send an email to the project owner, stating the need for remediation work on the inventory item.
	
	• **automatically create a remediation task**—Automatically create a remediation task assigned to the default development contact (for example, an engineering manager), and send an email, notifying the contact about the assigned task. (The default development contact is defined at the project level, as described in "Updating Inventory Review and Remediation Settings for a Project" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.)
	
	• **automatically create a remediation task and an external work item**—Automatically do the following:
	
	  ◦ Create a remediation task assigned to the default development contact (for example, an engineering manager), and send an email, notifying the contact about the assigned task. (See the previous bulleted item for more information.)
	
	  ◦ Associate a work item with the task, creating the work item in an Application Lifecycle Management (ALM) system (such as an issue in Jira). The work item is created and assigned using the settings defined for the ALM instance to which the Code Insight project is associated. For more information about configuring an ALM instance for the project, see "ALM Settings" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.

About FlexNet Code Insight Server REST APIs

You can create an administration client (tool) that communicates with the FlexNet Code Insight server using REST APIs to manage scan operations and to retrieve inventory information. These APIs use a REST-style interface and JSON. For more information about the Rest APIs, see the *Rest API Guide* Swagger documentation available from the **Help** menu.

Task ***To view REST API documentation, do the following:***

1. From any page in FlexNet Code Insight, click  and select **Help** from the menu. The **Documentation** menu appears.
2. Click **Rest API Guide**. The REST API documentation appears in a tab in your browser.
3. To view details about a particular item, click the arrow (>) next to the item. Additional information, if available, appears under the selected item.
4. (Optional) With the details about the API visible, click the API type (GET, POST). More information about the API appears. Click **Try it out** and then click **Execute**. The application will generate cURL, make the Rest API call and display a response.

3

Integrating with Source Code Management

The following topics are covered in this section:

Why Use Source Code Management (SCM)?

To support deep scanning, it is necessary to bring the project codebase files to the scan server. FlexNet Code Insight provides the following ways to bring codebase files into the system:

- **Upload a codebase into FlexNet Code Insight**: Uploading a codebase is useful to analysts who typically perform ad-hoc scans on an arbitrary snapshot of code provided by the product team.
- **Use a version control SCM connector**: SCM connectors provide an automated way to fetch the code based on criteria, such as build, release, calendar, checkin, and other information. SCM connectors support various authentication mechanisms, including anonymous, username and password, and token, key, or ticket on a scan server.

  See the next section, SCM Support, for information about the SCM systems for which FlexNet Code Insight provides connector support.

SCM Support

FlexNet Code Insight provides connector support for the following SCM systems, enabling remote codebases in these systems to be obtained before a scan:

- GIT
- Perforce
- TFS

The next sections describe the prerequisites that need to be in place before the Code Insight scan server integrate with any of the supported SCM systems:

- SCM Command-Line Client
- Git Protocol Configuration
- Perforce Protocol Configuration
- TFS Protocol and Credentials Configuration

For information about configuring a synchronization instance to a specific codebase in the SCM system, see the "Configuring Source Code Management" chapter in the *FlexNet Code Insight User Guide*.

SCM Command-Line Client

Before you proceed, ensure that an SCM command-line client is installed and configured on the FlexNet Code Insight scan server as this is necessary for FlexNet Code Insight to be able to connect to and synchronize with an SCM repository. See the following topics for information:

- Recommended Clients
- Verifying SCM Client Installation
- Setting the Environment Variable on Windows
- Prerequisite If Running Code Insight as a Service

Recommended Clients

The following is a list of clients known to work well with FlexNet Code Insight:

SCM	Client	Download Site
Git	Git	http://git-scm.com/downloads
Perforce	Perforce	https://www.perforce.com/downloads
Team Foundation Server (TFS)	Team Explore Everywhere Command Line Client (TEE-CLC)	https://github.com/Microsoft/team-explorer-everywhere/releases

Note • *Download site links are subject to change.*

TEE-CLC Requirement for a TFS Connection

TEE-CLC is the TFS client required by Code Insight to connect to and synchronize with an TFS collection. Once this client is installed on the same machine where the Code Insight scanner resides, run the following command to accept the end-user license agreement:

```
tfs -eula
```

If Code Insight attempts to connect to TFS before this command is run, the connection fails.

Verifying SCM Client Installation

To verify that the SCM client is installed and available to FlexNet Code Insight, open a command prompt and navigate to the FlexNet Code Insight root directory. Execute a command specific to your SCM, such as:

- `git help`
- `p4 help`
- `tf help`

If the system cannot find the command specified, verify that the SCM client directory is part of the PATH variable on this server. Consult your SCM documentation for more information on how to install and configure the client.

Additionally, best practice is to actually check out a sample repository on the SCM server to ensure that connectivity between the SCM client and SCM server is configured appropriately.

Setting the Environment Variable on Windows

If you run the SCM command line client from a Windows machine, add your SCM client location to the PATH environment variable.

Note • *Your SCM may require other environment variables to be set. Consult your SCM documentation.*

Task ***To set the environment variable, do the following:***

1. To find your PATH environment variable settings, navigate to **Control Panel > System > Advanced System Settings**.
2. Click **Environment Variables**.
3. Look for the PATH system variable and make sure that it is set to the location of your SCM bin directory.
4. If you edit the system variable, ensure that you save your changes.

Prerequisite If Running Code Insight as a Service

If Code Insight is configured to run as a service, the user context under which the service runs must have the appropriate permissions to run the SCM client.

Git Protocol Configuration

Git repositories reside on public servers, such as GitHub and Bitbucket, or on Git servers within a corporate network. The Git URL used to clone the repository into your SCM destination folder will vary depending on your desired protocol. The following are the available protocol options. You will enter then enter the URL using the desired protocol when you configure the SCM instance in Code Insight to connect to repository you want to clone in Code Insight.

- Anonymous HTTP
- Authenticated HTTP
- HTTPS
- SSH

Note • *Ensure that you are able to run the Git client on the machine where FlexNet Code Inside resides.*

Anonymous HTTP

This protocol can be used for a public repository. Public repositories can be cloned without providing an account and password.

Type	Example
GitHub Example	`http://github.com/myacct/Spoon-Knife.git`
Bitbucket Example	`http://bitbucket.org/myacct/myquotefork.git`

Authenticated HTTP

This protocol can be used for a private repository. Provide an account and password as shown in the URL format below. Use a colon between the account and password.

Type	Example
GitHub Example	`http://myacct:password@github.com/myacct/Hello-World.git`
Bitbucket Example	`http://myacct:password@bitbucket.org/myacct/bb101repo.git`

HTTPS

FlexNet Code Insight supports an anonymous or authenticated HTTPS protocol between a system running Code Insight and Git servers such as GitHub and Bitbucket.

HTTPS Configuration

Ensure that the SSL certificate verification between the Git client, installed on the Code Insight server, and the Git server is successful. This verification might include importing the Git server certificate into the local cacert authority. Because the details of this process is outside the scope of FlexNet Code Insight documentation, refer to the appropriate GIT server or client documentation for further details.

In a trusted environment, one option to ease the integration process is to skip the SSL certificate validation step. You can use the following command, run on the Code Insight server on which the Git client is installed, to skip the SSL certificate verification:

```
git config --global http.sslVerify false
```

URL Format

For an anonymous HTTPS protocol, use a URL similar to one of these:

Type	Example
GitHub Example	`https://github.com/myacct/Spoon-Knife.git`
Bitbucket Example	`https://bitbucket.org/myacct/myquotefork.git`

For authenticated HTTPS protocol, provide an account and password, separating them with a colon as shown in these examples:

Type	Example
GitHub Example	`https://myacct:password@github.com/myacct/Hello-World.git`
Bitbucket Example	`https://myacct:password@bitbucket.org/myacct/bb101repo.git`

SSH

FlexNet Code Insight supports SSH authentication between a system running Code Insight and Git servers such as GitHub and Bitbucket. Refer to the GitHub documentation (such as https://help.github.com/articles/connecting-to-github-with-ssh/) for details about this setup.

Perforce Protocol Configuration

Perforce depots reside on an enterprise server. The following protocol options are supported:

- Authenticated TCP
- Authenticated SSL

TFS Protocol and Credentials Configuration

The following describes configuration you might need for Code Insight synchronization with TFS:

- HTTPS Protocol Support
- Special Requirement for VSTS Projects in TFS

HTTPS Protocol Support

HTTPS is supported for communication between Code Insight and TFS. Perform the following steps to enable the SSL configuration for HTTPS.

Task ***To enable SSL configuration, do the following:***

1. Export the Secure Site SSL certificate from the browser location (shown here) for the given TFS instance:

   `https://<TFS-Host>/tfs/DefaultCollection/<Project>`

2. Import the certificate in the Java (JRE) keystore, using the following command (replacing `tfs.cer` with the actual certificate file name). The certificate should be imported to the same location where the TEE-CLC and Code Insight scanner reside (see TEE-CLC Requirement for a TFS Connection).

   ```
   keytool -import -trustcacerts -keystore cacerts -storepass changeit -noprompt -alias tfs -file
       tfs.cer
   ```

Special Requirement for VSTS Projects in TFS

If Code Insight is synchronizing with a VSTS (Visual Studio Team Services) project in TFS, alternate VSTS authentication credentials are required for the synchronization.

Task ***To enable alternate authentication credentials needed for Code Insight synchronization with a VSTS project in TFS, do the following:***

1. In Visual Studio, enable a set of alternate authentication credentials. (See the Visual Studio documentation for instructions.)
2. Specify these alternate credentials for the **Username** and **Password** in the TFS SCM instance configuration in Code Insight. See “Adding a TFS SCM Instance to the Code Insight Project” in the “Configuring Source Code Management” chapter in the *FlexNet Code Insight User Guide*.

4

Integrating with Application Lifecycle Management

This chapter covers the following topics:

About Integration with Application Lifecycle Management (ALM) Systems

FlexNet Code Insight integrates with application lifecycle management (ALM) systems, enabling Code Insight users to create and manage external work items associated with inventory directly from Code Insight. In this way, inventory requiring further review and remediation can be tracked externally as part of the user's existing issue-tracking system.

For example, a Code Insight scan might uncover security vulnerabilities or copyleft licenses requiring further review by the Security and Legal teams. With an ALM integration, these issues can be quickly converted into work items that point to corresponding issues in the ALM instance.

Integration with a specific ALM system is enabled through a corresponding Code Insight connector that supports pre-populated data (in the form of one or more ALM *instances*) used to connect to the ALM system and to set up work items. Additionally, a given ALM instance controls the synchronization of data between Code Insight and the server based on a configured synchronization frequency. To configure an ALM connector, the Code Insight Administrator defines one or more of these instances in Code Insight, as described in this chapter.

A given project can then be associated with one of the instances, enabling project integration with the ALM system so that users can create and manage the project's work items. (See "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide* for a description of this process.

Currently, Code Insight provides a Jira connector only (see the next section, The Jira Connector). Future releases will provide additional integrations with other ALM systems.

The Jira Connector

The Jira connector provided by Code Insight can be used to create new Jira work items directly from Code Insight. These work items allow management of external remediation work associated with inventory items in Code Insight.

The following sections describe how to configure the Jira connector for Code Insight integration with your Jira instances:

- Prerequisites for the Jira Connector
- Configuring the Jira Connector

Prerequisites for the Jira Connector

The Jira connector is included with Code Insight, is located on the core server in the `config/core/plugins` directory. Ensure that this directory contains the latest Jira connector, particularly after migrating to the latest Code Insight version.

Additional requirements include the following:

- The Jira connector requires access to a Jira server with credentials for a valid user on this server. The designated user will be used to authenticate Code Insight on the Jira server and will also be listed as the reporter on the issue created from Code Insight.
- The specified use must have full access to the Jira instance, particularly if Captcha or Single Sign-On are enabled on the Jira server.

You can use the **Test Connection** button on the ALM configuration page for the Jira instance to validate a successful connection to the Jira server. (See Adding a Jira Instance in the next section, *Configuring the Jira Connector*.)

Configuring the Jira Connector

The Jira connector can be configured to connect to multiple Jira instances and to display default values for each field in the configured instance. Projects can then be individually assigned to connect to and synchronize to one the configured instances.

The following topics describe how to configure and maintain a Jira instance:

- Adding a Jira Instance
- Using Code Insight Variables
- Synchronizing Work Items
- Deleting an ALM Instance

Adding a Jira Instance

The system Administrator can configure one or more Jira instances and their default field values globally at the application level using the **Administration** menu. Once configured, the Jira instances are available in the **Edit Project** section so that they can be associated to a specific project.

Task ***To add a Jira instance, do the following:***

1. As system Administrator, select **Administration** from the main menu.
2. Select the **ALM** tile on the left.
3. Select **Jira** from the **Application** dropdown list.
4. Click **Add Instance**. The **Instance** configuration tab is displayed.
5. Enter values for the required fields based on your Jira server information. The following fields are required. (See the inline help for explanations of the fields.)
   - **ALM Instance Name**
   - **JIRA Server URL**
   - **JIRA Username**
   - **JIRA Password**
6. Once you have completed the required fields, click the **Test Connection** button on the right to validate that Code Insight can connect to the specified Jira server.

   If the connection is successful, a "connection successful" message is displayed. Otherwise, reenter the credentials and try again. Ensure that the specified user has full access to the Jira instance, particularly if Captcha or Single Sign-On are enabled on this Jira server.
7. Complete the remaining fields. See the inline help for explanations of the fields.

   You can include inventory variables in the **Default Summary** and **Default Description** fields that will be replaced by actual values in the newly created Jira issue and work item. For a list of supported variables, see the next section, Using Code Insight Variables.
8. Click **Save** to save the Jira instance. The Jira sever settings and mandatory values are validated.

Using Code Insight Variables

The **Default Summary Text** and **Default Description Text** fields support Code Insight variables that can communicate details about the Code Insight project, inventory item, and other relevant information in the work item and associated Jira issue.

Supported Variables

The following table lists the available variables for use in the text entered in the **Default Summary Text** and **Default Description Text** fields:

Table 4-1 • Supported Code Insight Variables For Use in Work Item Summary and Description Text

$PROJECT_NAME	Name of the Code Insight project containing the issue
$INVENTORY_ITEM_NAME	Name of the inventory item containing the issue

Table 4-1 • Supported Code Insight Variables For Use in Work Item Summary and Description Text (cont.)

$COMPONENT_NAME	Name of the component associated with the inventory item
$VERSION_NAME	Version of the component associated with the inventory item
$LICENSE_NAME	Name of the selected license for the inventory item
$NUMBER_VULNERABILITIES	Total number of security vulnerabilities associated with the inventory item
$NUMBER_FILES	Total number of files associated with the inventory item
$INVENTORY_URL	Link to the inventory item

When the work item is created, the included variables are replaced by their respective values.

Example Use of Variables

The following is example text you might enter in the **Default Summary Text** field. The text includes some of the available variables:

```
The $INVENTORY_ITEM_NAME inventory item in the project $PROJECT_NAME contains
$NUMBER_VULNERABILITIES vulnerabilities that require review. Go to $INVENTORY_URL to see the
vulnerable inventory item.
```

If your Code Insight project name is *MySampleProject* and the name of the inventory item name for which you create a work item is *Apache Commons BeanUtils*, the work item and Jira issue will display the following summary:

```
The Apache Commons BeanUtils 1.7.0 (Apache 2.0) inventory item in the project MySampleProject
contains 18 vulnerabilities that require review. Go to https://my.sample.server:8888/codeinsight to
see the vulnerable inventory item
```

Synchronizing Work Items

FlexNet Code Insight provides the ability to synchronize work items between Code Insight and the ALM system so that Code Insight always reflects the most current state of each work item. The one-way synchronization updates the following fields for the work item in Code Insight: **Status**, **Type**, **Priority**, **Assignee**, **Summary**.

The following procedure describes how to set the frequency of this synchronization process (labeled **Existing Issues Sync Frequency** on the ALM tab).

Note *• The Sync Frequency configuration applies to all the ALM instances. If not explicitly set, the sync frequency defaults to Daily.*

Task ***To configure the issue sync frequency, do the following:***

1. As system Administrator, select **Administration** from the main menu.
2. Click the **ALM** tab.
3. Click the **Edit Sync Frequency** icon on the right (to the right of the **Existing Issues Sync Frequency** value).
4. Select one of the frequency options—**Never**, **Hourly**, **Daily**, or **Weekly**—and complete their respective sub-options.
5. Click the **Save Changes** icon to save or **Cancel** to discard the setting.

Work Item Status Updates

If the status of the work item in the ALM system changes, the status of the work item in Code Insight will reflect the change after the synchronization completes.This can result in a change to the **# Open Work Items** and **# Closed Work Items** for each inventory item. These links and the **Open Work Items** information alert link will be updated to reflect the change. Additionally, the **Inventory with Open Work Items** selection in **Advanced Search** may return a different number of results.

The following lists the default status values:

- The default Open status values for Jira include **Open**, **Reopen**, **New**, **To Do**, **In Progress**, and **Backlog**.
- The default Closed status values for Jira include **Done**, **Resolved**, **Verified**, and **Closed**.

Custom statuses are not currently supported.

Deleting an ALM Instance

The application Administrator can delete an ALM instance as long as no projects currently reference the instance.

If the instance that you want to delete is referenced by a project, it cannot be deleted until the instance is unassociated from the project. See the *FlexNet Code Insight User Guide* for instructions on how unassociate an instance from a project.

Task ***To delete an ALM instance, do the following:***

1. As the system Administrator, select **Administration** from the main menu.
2. Select the **ALM** tab.
3. Select the **Instance** tab for the instance you want to delete.
4. Click the **Delete Instance** button.

A

FlexNet Code Insight User Roles and Permissions

This appendix serves as a reference to the various user roles and permissions that Code Insight offers as a means to control user access to its functionality at your site:

- System Roles and Permissions
- Project Roles and Permissions
- Roles and Permissions to Manage Project Task Flow

System Roles and Permissions

The following table lists the system roles and associated permissions used to manage the FlexNet Code Insight system. The initial FlexNet Code Administrator (and any subsequent administrators) assigns these roles to other FlexNet Code Insight users by using the **Manage Permissions** dialog accessed from the Administration **Users/Permissions** tab. (For details, see Managing User Permissions for System Activities in the "Configuring FlexNet Code Insight" chapter.)

One user can be assigned multiple roles.

For Information on creating and managing FlexNet Code Insight users in general, see Managing Users in the "Configuring FlexNet Code Insight" chapter.

Table A-1 • System Roles and Permissions

		Roles		
		Administrators	Manage Policy	Create Project
Permissions	**Notes**			
Administer Code Insight: • Manage users and their roles and permissions • Schedule or run electronic updates • Configure an email server • Configure LDAP users • Configure Application Lifecycle (ALM) instances • Configure a scan server • Configure scan profiles • Define global project defaults		✔	–	–
Manage policies: • Create and edit policy profiles		–	✔	–
Create projects: • Create projects (and thereby automatically become Project Owner for each) • Create project folders (in **Projects** list)	The Project Creation role is controlled by the **Allow All Users to Create Projects** option on the **Manage Permissions** dialog. If **Yes** (default), any user has this role. If **No**, only users assigned this role can create projects. (For details, see Managing User Permissions for System Activities in the "Configuring FlexNet Code Insight" chapter.) Users who have this role can also create project folders in the **Projects** list.	–	–	✔

Project Roles and Permissions

The following table lists the various roles and associated permissions used to manage a given project in Code Insight. The Project Owner assigns the Analyst, Reviewer, and Observer roles to FlexNet Code Insight user and can reassign project ownership. For details about these roles and the procedure for assigning them, see "Assigning Project Roles to Users" in the "Using FlexNet Code Insight" chapter in the *FlexNet Code Insight User Guide*.

Table A-2 • Project Roles and Permissions

		Roles			
		Project Owner	Analyst	Reviewer	Observer
Permissions	**Notes**				
Manage project: • Change project owner • Manage project users • Rename project • Move projects in Project Folder Tree • Manage scan settings • Manage and inventory review/ remediation settings • Manage Source Control Management (SCM) and Application Lifecycle (ALM) instances	The project creator automatically becomes Project Owner, who can then reassign ownership to another user. See the previous section, System Roles and Permissions, for information about the Create Project role needed to create projects.	✔	–	–	–
View project inventory	Any user (not just one with a project role) can view the **Project Inventory** tab and the associated inventory details.	✔	✔	✔	✔*
Edit/create project inventory	The Reviewer role has limited inventory-editing capabilities on the **Project Inventory** tab. Reviewers can neither edit nor add new inventory (that is, the **Edit** and **Add** buttons are not available). However, reviewers can recall inventory and edit inventory priority, review status, alerts, as-found license text, and notes (except detection notes).	✔	✔	✔	–
Access Analysis Workbench • View codebase file tree • Edit inventory in the Workbench		✔	✔	–	–

Table A-2 • Project Roles and Permissions (cont.)

		Roles			
		Project Owner	Analyst	Reviewer	Observer
Permissions	**Notes**				
Generate reports	Any user (not just one with a project role) can generate reports. For a "private" project, the Observer is considered a regular user of the project, restricted to viewing project inventory and generating reports.	✔	✔	✔	✔*
Invoke a scan		✔	✔	—	—
Upload codebase		✔	✔	—	—
Import/export project data		✔	✔	—	—

* The Observer role is available for only projects defined as "Private". Only Observers, the Project Owner, Analysts, and Reviewers have access to the "Private" project to which they are assigned. The Observer is considered a regular user, restricted to viewing project inventory and generating reports for the "Private Project".

Roles and Permissions to Manage Project Task Flow

The following table lists the project roles and permissions used to manage tasks to review or remediate inventory items in a project.

Table A-3 • Project Task-Flow Roles and Permissions

		Roles				
		Project Owner	Analyst	Reviewer	Observer	Task Assignee
Permissions	**Notes**					
Create/edit tasks	Any user assigned to a project role can create and edit tasks.	✔	✔	✔	✔	✔
Reassign task		✔	—	—	—	✔
Close manual review task		—	—	✔	—	—
Close remediation task		✔	—	—	—	✔

Table A-3 • Project Task-Flow Roles and Permissions (cont.)

		Roles				
		Project Owner	Analyst	Reviewer	Observer	Task Assignee
Permissions	**Notes**					
Close miscellaneous task	Any user assigned to a project role can close a miscellaneous task.	✔	✔	✔	✔	✔