lgtm enterprise

LGTM Enterprise
Installation and Upgrade Guide

Release 1.20.2, May 2019

Semmle™

Semmle Inc.
44 Montgomery Street
3rd Floor
San Francisco, CA 94104

LGTM Enterprise release 1.20.2
Document published May 14, 2019

Contents

Introduction

What is LGTM?

LGTM is a variant analysis platform that combines deep semantic code search with data science insights to help developers ship secure code.

> **Note**
> "LGTM" is the most commonly posted comment on code review platforms (such as GitHub). It means: "looks good to me."

About LGTM Enterprise

LGTM Enterprise is built on Semmle's tried and tested analysis technology. The product is the on-premises twin of Semmle's LGTM.com platform—a publicly available, free to use, cloud-based service that continuously analyzes over 135,000 open source projects worked on by over 700,000 developers, and containing data for more than 39 million commits.

Most notably, LGTM Enterprise offers the full power of LGTM variant analysis to customers in an on-premises version. LGTM allows developers to efficiently browse alerts in their source code (using either a web browser or a plugin for their IDE), while at the same time incorporating broader insights. It seamlessly integrates with platforms like GitHub Enterprise and Bitbucket Server for code review, authorization, and authentication.

About this document

This guide provides instructions for installing and upgrading LGTM Enterprise.

Two methods are described:

- **Interactive deployment**

  This involves running the supplied installation script, or upgrade script, and entering information in response to prompts.

By default, the installation script installs LGTM Enterprise on a single machine and runs just one general worker daemon to perform project analysis. It's therefore not suitable for analyzing many or large projects. It can, however, be used for demonstration or simple evaluation purposes. It also provides the basis for a larger deployment, by giving you a simple system to which you can add worker host machines, to provide additional processing resources. You can find information on expanding your system in LGTM's administrator help.

- **Programmatic deployment**

  This method is designed for system administrators who want to be able to install or upgrade LGTM Enterprise on machines in an easily repeatable way, without having to step through prompts for information, entering details manually.

Upgrading LGTM to the latest version

The upgrade process is very similar to the installation process. On a distributed LGTM cluster, upgrades must be performed on each machine in the cluster. The instructions in this guide cover upgrading your system.

Before upgrading, check the "Appendix—Notes for upgrading LGTM Enterprise" on page 51 for any pre-upgrade steps you need to take.

Time required

You should allow an hour to perform an initial test installation of a basic, single-machine system. Installing and setting up LGTM Enterprise on a single virtual machine is a good way to start if you have not performed an installation before. Installing a larger, multi-machine system for live use will take proportionally longer, depending on the architecture you choose.

After installing LGTM Enterprise you should check the system by adding some projects for analysis (you can delete these later, if required). You need to allow the application time to perform the analysis. If you're installing an initial, temporary instance of LGTM Enterprise, it's a good idea to add just a few projects at first, and come back in three or four hours to check that results that are available on the site. Alternatively, you can add more projects and leave the application to run overnight before looking at the analysis results.

Upgrade is a simple process and, in most cases, should not take more than an hour (excluding any pre- and post-upgrade steps that may be required).

Related documentation

- *LGTM Enterprise System Architecture* (PDF)
- *LGTM Enterprise System Requirements* (PDF)
- LGTM Enterprise administrator help

  This is available at help.semmle.com/lgtm-enterprise/admin, or by clicking **Admin help** at the top of the administration pages in LGTM Enterprise.

Prerequisites for installation

The LGTM Enterprise distribution is designed to make installation and configuration as easy as possible. It includes almost all of the components you need. For example, the packages that are installed include the nginx HTTP and reverse proxy server for serving pages from the LGTM Enterprise web application. Some prerequisites are, however, assumed and it is important that you make sure these are all in place before proceeding with installation.

The instructions in this document assume that all of the following requirements have been met:

- You have a valid Semmle license for LGTM Enterprise.

- You have familiarized yourself with the architecture of an LGTM Enterprise system and decided on a specific architecture for your installation. For information, see the *LGTM Enterprise System Architecture* PDF, or the topics under "Server overview" in the LGTM Enterprise administrator help at https://help.semmle.com/lgtm-enterprise/admin/help/sys-architecture/server-overview.html.

  For example, a demo system can be contained on a single machine. A trial system might host all of the control pool components on one machine and use one or more additional machines for the work pool. A live deployment of a large system might use several more machines for a fully distributed cluster design.

- The machines that are going to host LGTM Enterprise are running a supported operating system. See the details in the *LGTM Enterprise System Requirements* PDF, or the administrator help.

  We recommend that you start with a fresh Linux installation (for example, on a virtual machine).

- The machines that are going to host LGTM Enterprise are appropriately resourced.

  For example, for a single-machine installation for demo or simple evaluation purposes, the machine should have a minimum of 8 cores, 16 GB of RAM, and 100 GB of free disk space.

  > **Important**
  > For a live deployment of LGTM Enterprise see the separate *LGTM Enterprise System Requirements* PDF, or the "System requirements" topic in the administrator help.

- On machine that is going to host the LGTM web application, port 443 is free and accessible remotely. For more information about port usage, see the *LGTM Enterprise System*

Architecture PDF, or Network connections in the administrator help.

> **Note**
> Alternatively, you can configure the bundled third-party nginx web proxy to use a different port for SSL connections. However, this is beyond the scope of this document. For more information, see the nginx documentation.

- The installing user must be able to run commands using sudo on all LGTM host machines.

Worker prerequisites

To analyze projects with LGTM, the machines on which worker daemons will build and analyze projects must have all of the environment prerequisites to allow this to be done. These prerequisites are likely to vary from project to project.

You can install the prerequisites for the projects you plan to build at any time, however, it’s usually simplest to install LGTM Enterprise first, then check the environment (see the topic "Configuring hosts to analyze code" in the administrator help), and then add any missing prerequisites. On Linux computers, this ensures that EPEL (Extra Packages for Enterprise Linux) is present and can be used to install the missing packages.

For full system requirement details for a live deployment of LGTM Enterprise, see the separate *LGTM Enterprise System Requirements* PDF (or you can find the same information in the administrator help).

SSL certificates

To ensure that data transferred to and from servers is protected, LGTM Enterprise relies on SSL/TLS encryption. When you install LGTM Enterprise, self-signed certificates are automatically generated to protect communication between internal components, for example, the web application and the file store.

By default, the web application is configured to use a self-signed certificate to communicate externally. Using this certificate can be useful initially for testing purposes. However, while you can establish trust using the self-signed certificate, this requires you to disable SSL verification in your repository management system. This is not recommended for production systems and corporate policies usually prevent this. Consequently, you require a true, trusted certificate for the domain at which the LGTM Enterprise web application will be located.

If you *do* decide to use the default self-signed certificate, be aware that:

- Pull request integration with will fail unless you carry out additional configuration on the repository system to permit self-signed certificates.
- Users will see a browser security message when they attempt to access LGTM Enterprise. They may need to add a security exception for the site before they can view it (behavior varies for different browsers).
- Users will be unable to use the LGTM plugins to view alerts in their IDE.

Important

Before installing LGTM Enterprise, obtain an SSL certificate and private key for the domain at which the LGTM Enterprise web application will be located. These should be in separate files, in PEM format. The certificate must be issued by a trusted certification authority and must *not* require a password to be supplied.

For more information about SSL certificates for LGTM Enterprise see "Certificate details" in the LGTM administrator help on help.semmle.com.

Third-party software installed by LGTM

LGTM Enterprise uses the following third-party software. If these resources are not found on the relevant machines during installation of LGTM Enterprise, they are automatically installed.

- Java JRE 8 (if necessary, OpenJDK 8 will be installed)
- Python 2
- PostgreSQL 9.5 or later
- RabbitMQ Server
- Nginx
- Git
- Subversion
- Solr 7.5.0 (bundled with LGTM)
- Minio release 2019-02-14T00-21-45Z (bundled with LGTM)

With the exception of Solr and Minio (which are supplied within the LGTM distribution), the version number of the software that's installed depends on what the latest available version is when the installation is performed. Any dependencies are also installed.

Access to third-party packages during installation

The simplest way to install LGTM Enterprise is to enable internet access and to run the installation script supplied with the application. This enables the installer to download and install the required packages from their official online package repositories using the standard utilities available in your Linux distribution. If you're using Red Hat Enterprise or CentOS, this will require access to the EPEL (Extra Packages for Enterprise Linux) repository. Once LGTM has been deployed, internet access can be closed down.

If internet access is not available, you can do either of the following:

- Configure the machine so that it can access an internally hosted mirror that contains these package repositories and their dependencies. The installer will download and install the required packages from your internal mirror.

- *For installations on Red Hat or CentOS:* Extract the third-party dependencies pack before you run the installer. Details for doing this are included in the installation instructions.

Installation and upgrade methods

This guide describes two methods of installing or upgrading LGTM Enterprise:

- **Interactive deployment**

  This is the simplest way to install or upgrade LGTM Enterprise. Just unpack the distribution and run the supplied installation script or upgrade script. You will be prompted to enter information about your system.

  To use this method, see: "Interactive deployment" on the next page.

- **Programmatic deployment**

  This method provides a way to store information about the system you want to create, so that you can install or upgrade LGTM Enterprise repeatedly, without having to manually enter replies to prompts. More initial preparation is required—to parameterize information and set up the means of issuing commands (either using a configuration management system, such as Ansible, or as a shell script)—but having done this, installing and upgrading LGTM Enterprise on virtual machines becomes a quick and easy to repeat process.

  To use this method, see: "Programmatic deployment" on page 24.

Interactive deployment

LGTM Enterprise comes with scripts that step you through the installation and upgrade processes. These scripts provide the easiest way to perform a one-off installation or upgrade of LGTM Enterprise. This section of the guide tells you how to use this interactive method of installing or upgrading LGTM Enterprise.

If, however, you need to perform repeated installations or upgrades with parameterized input that does not require you to type in answers to prompts for information, then you'll probably want to consider configuring programmatic installation or upgrade for LGTM Enterprise—see "Programmatic deployment" on page 24.

Installation overview

If you are performing an installation, the instructions given here assume that you have already set up one or more machines on which to install LGTM Enterprise See the "Prerequisites for installation" on page 9.

Tip
Typically you will be installing onto virtual machines (VMs) and it's strongly recommended that you take a snapshot of the VMs before starting the installation, so that you can easily start the process over if required.

Deployment phases

The interactive deployment method comprises two main phases:

1. **Preparing to install or upgrade**

   Uploading the license file, SSL certificate and distribution file, then unpackaging the distribution file.

   See "Preparing to install or upgrade" on the next page.

2. **Running the installer or upgrader**

   See:

"Installing LGTM (interactive method)" on the next page

"Upgrading LGTM (interactive method)" on page 22

After the installation or upgrade script finishes you will log into LGTM Enterprise and perform some checks. For an installation, you can then start to configure your system as required. Links to useful help topics on these subjects are provided later in this guide.

Preparing to install or upgrade

Before using the interactive scripts to install or upgrade LGTM Enterprise, you need to perform the following tasks.

Upload and extract the LGTM distribution

1. Open a command console for the machine that will host the LGTM coordinator (part of the control pool).

   For example, connect to the server using an SSH client.

2. Create a directory to store installation and configuration files used during an installation or upgrade. In this guide, the commands use an example directory of `lgtm-releases`, at the `$HOME` location, but you can use any directory for this purpose. For example:

   ```
   mkdir $HOME/lgtm-releases
   ```

   > **Note**
   > The files in this location are not needed after the installation or upgrade is complete.

3. Upload the distribution file for LGTM Enterprise (`lgtm-<version>.tar.gz`) into the `lgtm-releases` directory.

4. *For Red Hat and CentOS only:* If internet access is not available and you want to perform an offline installation or upgrade, upload the supplied `lgtm-third-party-rpms-<yyyy-mm-dd>.tar.gz` file into the same directory.

5. In the command console, change to the `lgtm-releases` directory and extract the contents of the LGTM distribution package by entering:

   ```
   tar -xvf lgtm-<version>.tar.gz
   ```

The `lgtm-releases` directory now contains an `lgtm-<version>` subdirectory, within which are subdirectories called generated and `lgtm`.

6. *For Red Hat and CentOS only:* If you're performing an offline installation or upgrade, extract the contents of the dependencies package into the `lgtm-<version>/lgtm` directory:

   ```
   tar -xvf lgtm-third-party-rpms-<yyyy-mm-dd>.tar.gz -C lgtm-<version>/lgtm
   ```

   This adds the directories: `database`, `queue`, and `web`.

Upload a license file and SSL certifcate

1. Upload your LGTM license file to the `lgtm-releases/lgtm-<version>/lgtm` directory which you created previously.

2. Change the file name to `license.dat` if it is not already called this.

3. Within the `lgtm-releases` directory, create a subdirectory called `state`.

4. Checkout copies of the SSL certificate and key file for the LGTM web application, from your version control system, into the `lgtm-releases/state` directory.

Install or upgrade?

You're now ready to run the installation or upgrade script and answer the prompts.

Continue with the appropriate section for an installation or an upgrade:

- "Installing LGTM (interactive method)" below
- "Upgrading LGTM (interactive method)" on page 22

Installing LGTM (interactive method)

The interactive installation script is stored in the `lgtm` subdirectory of the `lgtm-<version>` directory that you extracted earlier, for example: `lgtm-releases/lgtm-<version>/lgtm/install.sh`. Typically, you will need to use the sudo command to run the script—this grants the elevated permissions that are needed to make the required file system changes.

Run the installation script

From the `lgtm-releases/lgtm-<version>/lgtm` directory enter:

```
sudo ./install.sh
```

Reply to the prompts, as follows:

1. `What is the fully qualified domain name of this machine (this will be used for internal communication between different cluster machines)?`

   The text in square brackets gives the fully qualified domain name (FQDN) of the current machine. Press Enter to use this name, or provide an alternative FQDN.

   > **Important**
   > If the server has both private and public FQDNs, specify the private FQDN here.

2. `What URL will users use to access LGTM?`

   The text in square brackets gives a URL that uses the FQDN you just entered.

   If this is not the URL that users will use in their browsers to access LGTM, enter the correct URL. Otherwise press Enter to accept the default.

3. `A cluster configuration for a single machine installation has been created. Do you want to edit it manually before continuing?`

   The cluster configuration file specifies the hostnames of machines on which the various cluster components will run. It also specifies the number of worker daemons that should be run to build and analyze projects. In addition, you can specify a custom SSL certificate and key file for the LGTM web application to use. For more information about editing the configuration file, see the topic "Editing and deploying the cluster configuration" in the administrator help.

   If you have an SSL certificate and key file for LGTM to use—or if you want to configure a particular setup at this stage, rather than using the simple, single-machine setup provided by the installer—type `y` and press Enter to edit the file. Alternatively, press Enter to continue to the next step.

   > **Note**
   > Editing the cluster configuration at this stage is optional. You can modify the system

later if required. If you have not installed LGTM Enterprise before, you may want to use the default configuration for now, and then modify it later—for example, increasing the resources for analysis by adding more worker daemons.

If you entered y:

If a local text editor is available, the cluster configuration file is opened for editing. Otherwise, you're prompted to edit the `lgtm-cluster-config.yml` file manually.

a. In the cluster configuration file, find the `ssl` block.

b. Within this block, add entries for `certificate_path` and `key_path`, specifying the names of the files.

```
ssl:
  hsts: false
  certificate_path: <certificate file>
  key_path: <key file>
```

Note

The paths to these files are relative to the cluster configuration file. If you have followed the instructions given here, this file is located in the `lgtm-releases/state` directory. This is the same place that you saved the certificate file and the key file, so you can specify their location by just giving the file names, without a path.

c. Check that the configuration is still valid YAML—for example, by using `yamllint`. For more information, see yamllint.readthedocs.io.

d. Save your changes to the configuration and close the file.

e. Return to the console where the LGTM installer is running.

f. If necessary, press Enter to continue the installation process.

4. `Please provide a password for encrypting the installation manifest:`

The LGTM manifest file contains sensitive information (for example, authentication details) which is encrypted. You are prompted to enter a password for encrypting/decrypting this data. You will need this password whenever the manifest is changed, for example, when you upgrade LGTM.

> **Note**
> Defining a password is optional. If control of the credentials stored in the manifest is not a concern, or if the file is protected by some other means, then you do not need to enter a password.
>
> If you press `Enter` without giving a password, the manifest is encrypted with a fixed password which provides only a superficial degree of security.
>
> If you define a password, make sure you store it securely as you will need it to upgrade the system in future.

Enter a password for encrypting/decrypting the file. This is not displayed on the screen.

5. `If you want to create a new admin account, please enter the email address to be used for logging in now. Otherwise, leave blank.`

   Enter a valid email address.

   > **Important**
   > You should create an LGTM Enterprise administrator now, to allow you to log in to LGTM when the installation is complete. If you do not do so now you will need to create a user from the command line after installation. For details of how to do this, see the reference topics for the create-user and grant-admin actions in the LGTM Enterprise administrator help.

6. `And what should the password be?`

   Enter a password. This is not displayed on the screen. you're asked to confirm this password before the script continues.

   The installer will proceed with the installation for a few minutes before the next prompt is displayed.

The installer now carries out the installation, displaying messages about various services being started. By default, the LGTM work pool has two workers.

The script outputs messages similar to those shown below, and ends by prompting you to deploy LGTM Enterprise to any other machines in the LGTM cluster:

```
[info] Starting LGTM...
[info] Starting postgres-lgtm...
[success] Started postgres-lgtm.
[info] Starting minio...
[success] Started minio.
[info] Starting rabbitmq-server-lgtm...
[success] Started rabbitmq-server-lgtm.
[info] Starting solr-lgtm...
[success] Started solr-lgtm.
[info] Starting lgtm-scheduler...
Created symlink from /etc/systemd/system/multi-user.target.wants/lgtm-
scheduler.service to /usr/lib/systemd/system/lgtm-scheduler.service.
[success] Started lgtm-scheduler.
[info] Starting lgtm-jobdealer...
Created symlink from /etc/systemd/system/multi-user.target.wants/lgtm-
jobdealer.service to /usr/lib/systemd/system/lgtm-jobdealer.service.
[success] Started lgtm-jobdealer.
[info] Starting nginx-lgtm...
Created symlink from /etc/systemd/system/multi-user.target.wants/nginx-lgtm.service
to /usr/lib/systemd/system/nginx-lgtm.service.
[success] Started nginx-lgtm.
[info] Starting lgtm-webapp...
Created symlink from /etc/systemd/system/multi-user.target.wants/lgtm-webapp.service
to /usr/lib/systemd/system/lgtm-webapp.service.
[success] Started lgtm-webapp.
[info] Starting lgtm-worker@1...
Created symlink from /etc/systemd/system/multi-user.target.wants/lgtm-
worker@1.service to /usr/lib/systemd/system/lgtm-worker@.service.
[success] Started lgtm-worker@1.
[info] Starting lgtm-worker@2...
Created symlink from /etc/systemd/system/multi-user.target.wants/lgtm-
worker@2.service to /usr/lib/systemd/system/lgtm-worker@.service.
[success] Started lgtm-worker@2.
[success] All LGTM services on this machine have been started.
[prompt] If you have manually edited the cluster configuration file to introduce
additional machines, complete the installation by copying $HOME/lgtm-releases/lgtm-
<version> to each machine in the cluster and running the
'generated/<hostname>/install-machine.sh' script. Press Enter when complete...
```

If you edited the default cluster configuration to specify other hosts, in addition to the current machine, deploy LGTM Enterprise to these machines following the instructions reported by the script. When you've finished deploying LGTM to all the hosts in the cluster, press Enter.

```
[success] You can now access the LGTM admin panel in your browser by going to
https://<lgtm-domain>/admin/ and logging in with the email and password you supplied
```

```
earlier.
```

You're ready to log in to LGTM Enterprise and start configuring the system—see "Accessing the administrator interface" on page 48.

Tip

For information about deploying LGTM to other machines in the cluster, see the following topics in the administrator help:

- Adding and removing workers
- Moving control pool components

Upgrading LGTM (interactive method)

The interactive upgrade script is stored in the `lgtm` subdirectory of the `lgtm-<version>` directory that you extracted earlier, for example: `lgtm-releases/lgtm-<version>/lgtm/upgrade.sh`. Typically, you will need to use the sudo command to run the script—this grants the elevated permissions that are needed to make the required file system changes.

Before proceeding with a interactive upgrade, make sure you have:

- Performed the preparatory tasks described in "Preparing to install or upgrade" on page 15.
- Read the relevant notes for the upgrade you're performing—see "Appendix—Notes for upgrading LGTM Enterprise" on page 51.

Each LGTM Enterprise release has its own version-specific `upgrade.sh` script. When you're ready to upgrade, run the `upgrade.sh` script for the version you are upgrading to. Run this on the coordinator server, as described below. When you run the script, you will be prompted to perform any required actions on other servers in the LGTM cluster. For information about the servers in an LGTM cluster, see Server overview.

Important

- LGTM Enterprise will be unavailable during the upgrade. However, you don't need to take the system down yourself, the upgrade script does this for you.
- If LGTM's `manifest.xml` file was encrypted with a custom password when the system was installed, you need to know the password to upgrade the system (if you've lost the password, see Changing or resetting a manifest password).

Run the upgrade script

1. Open a command console for the coordinator server.

   Note

   The host name of the coordinator server is specified in the `<LGTM releases directory>/state/lgtm-cluster-config.yml` file—where *<LGTM releases directory>* is the directory into which the installation package for the current release was unpacked (for example, `$HOME/lgtm-releases/`).

2. Display the version of the currently running LGTM system by entering:

   ```
   sudo lgtm-cli version
   ```

3. Make a note of the version number. You will use this later to confirm that the system was upgraded.

4. Change to the `lgtm-<version>/lgtm` directory, created when you unpacked the distribution file.

5. Run the `upgrade.sh` script in this directory:

   ```
   sudo ./upgrade.sh
   ```

   Note

   - The script expects to find a cluster configuration file called `lgtm-cluster-config.yml` within a `state` directory located alongside the `lgtm-<version>` directory—for example, `$HOME/lgtm-releases/state/lgtm-cluster-config.yml`. If the cluster configuration file is located elsewhere, give the path to this file as an argument to the script:

     ```
     sudo ./upgrade.sh <alternative-path-to>/<cluster-configuration-file>
     ```

   - The script must be run using sudo, or as the root user.

6. Enter the password that was used to encrypt the manifest file when LGTM Enterprise was originally installed.

7. Follow the remaining on-screen prompts.

8. When the upgrade is complete, check the expiration date of your license by entering:

   ```
   sudo lgtm-cli license
   ```

   If there are less than three months left before expiry, we recommend that you make a reminder to upload a new license before this date to avoid disruption to users. If the license is due to expire soon and you haven't received a new license from your LGTM support team, contact them now. New licenses can be uploaded via the administration pages, or using the `license` CLI action, at any time.

Post-upgrade tasks

There are just a few final tasks to complete: see "Post-upgrade procedures" on page 50.

Programmatic deployment

Programmatic deployment is a method of installing or upgrading which, after it has been started, requires no human interaction. It will simply run through the entire process using details that have previously been stored in a script or a configuration management system. This allows you, for example, to run commands that will commission virtual machine instances, install LGTM Enterprise, and start the application in a ready-to-use state.

Installation or upgrade?

The installation and upgrade processes are very similar. The procedures described in the following sections cover both installing LGTM Enterprise from scratch, and upgrading an existing instance of LGTM Enterprise to a newer version. Sections, or individual steps, labeled *For installation only* or *For upgrade only* can be omitted depending on whether you are creating the command sequence for an installation or for an upgrade.

Procedural overview

Deployment consists of the following main stages:

1. *For installation only:* Create and provision virtual machines

2. Design and generate the cluster configuration

3. Install LGTM packages and configuration-related files on each machine in the cluster

4. Initialize/upgrade the LGTM database and start all components

Deployment procedures

After the initial stage in the programmatic deployment process (see "Creating and provisioning machines" on the next page) this guide describes procedures for two different cluster topologies. Choose which of these works best for you:

- A simple, **single-machine system** suitable for demo purposes

  Install or upgrade LGTM Enterprise entirely on one machine. We provide an example bash script that you can modify and run to perform a single-machine deployment. If required, you can save the same commands used in the script in a configuration management system—but running a script provides an easy way of performing a simple programmatic deployment.

See: "Single-machine programmatic deployment" on page 28.

- A **distributed system** where cluster components are hosted on multiple virtual machines

  Typically, a distributed cluster might host the core components (database, file storage, message queue, and search service) on one machine, together with the web application, and use a number of other machines to host worker daemons. Alternatively, you can also host the web application on one or more separate machines. The distribution of cluster components is very flexible. For more information, see the *LGTM Enterprise System Architecture* guide (PDF).

  The instructions for installing or upgrading a distributed cluster assume that you're using a configuration management system such as Ansible to store and replay commands.

  See: "Distributed cluster programmatic deployment" on page 34.

Creating and provisioning machines

The "Prerequisites for installation" on page 9 require you to have commissioned the machines on which you are going to install LGTM Enterprise. If you are using a configuration management system however, you could add the creation of virtual machines as the first step in your programmatic installation procedure.

Having commissioned appropriately resourced machines, configure your Ansible playbook (or other configuration management command sequence) to perform the following provisioning tasks.

Check for Java

For installation only

The interactive installer installs Java automatically if it is not found. However, for a programmatic installation, you need to make sure Java is available on the machines on which you're installing LGTM Enterprise

Check for Java, and install it if it is not found.

For example, run:

```
java -version
```

If the return value does not show that either OpenJDK or Oracle Java is installed (IBM Java is not supported), you must install it now:

- *On Debian/Ubuntu:*

  ```
  sudo apt-get update && sudo apt-get install openjdk-8-jre-headless
  ```

- *RHEL/CentOS:*

  ```
  sudo yum install java-1.8.0-openjdk-headless
  ```

Create a location for temporary LGTM files

Create a directory to store installation and configuration files used during an installation or upgrade. In this guide, the commands use an example directory of `lgtm-releases`, at the `$HOME` location, but you can use any directory for this purpose. For example:

```
mkdir $HOME/lgtm-releases
```

Note
The files in this location are not needed after the installation or upgrade is complete.

Create a location for version-non-specific files

Create a `state` directory to store files that are not supplied with LGTM, are not version-specific, and relate to the instance of LGTM Enterprise you are installing or upgrading:

```
mkdir $HOME/lgtm-releases/state
```

Note

- The `state` directory is intended as a temporary location in which to store copies of files that you will use in future deployments, such as the SSL certificate for the web server(s), and the system's manifest file. We recommend that you version control this directory. You can then checkout the `state` directory for each deployment of this system.

- The location of these files is not prescribed, and you can change it if required. However, locating the files in a directory called `state` is recommended for ease in following the documentation and discussing issues with Semmle.

Locate the certificate, license, and providers files

Do the following on the coordinator host machine:

1. Copy the following files to the `$HOME/lgtm-releases/state` directory:

   - The SSL certificate file for the domain of the URL at which users will access LGTM.

     In the examples given here this file is called `lgtm-web.crt`.

   - The corresponding SSL key file

     In the examples given here this file is called `lgtm-web.key`.

   > **Note**
   > For more information, see "SSL certificates" on page 10.

2. Copy your license file for LGTM Enterprise to any directory from where it can be loaded into LGTM during installation or upgrade (for example, your home directory on the LGTM coordinator host machine).

3. *Optional—for installation only:* If you have previously saved the provider settings for an LGTM Enterprise installation, put this file in a secure location that is accessible from the coordinator machine.

   > **Important**
   > The providers file contains secrets—for example, the passwords of accounts on third-party systems—so you should take appropriate care to protect this file.

Upload and unpack the LGTM distribution file

Upload the LGTM Enterprise distribution file (`lgtm-`*`<version>`*`.tar.gz`) to the `$HOME/lgtm-releases` directory of the coordinator host machine.

Change to the `lgtm-releases` directory, then extract the contents of the LGTM distribution package file:

```
tar -xvf lgtm-<version>.tar.gz
```

The `lgtm-releases` directory now contains an `lgtm-`*`<version>`* subdirectory, within which are subdirectories called generated and `lgtm`.

Single-machine programmatic deployment

All of the components of LGTM Enterprise can be run on a single machine. This can be useful to provide a very simple setup for demo purposes but the resulting system will lack the necessary resources to serve as a performant, multi-user system.

If you want to install or upgrade a multi-machine system, see "Distributed cluster programmatic deployment" on page 34.

Generate a cluster configuration file

For installation only

Prior to running a programmatic deployment of LGTM Enterprise for the first time, you need to create a file describing your single-machine "cluster."

1. Use the supplied cluster configuration generation tool to create a simple cluster configuration file.

   From the `lgtm-releases` directory, run:

   ```
   java -jar lgtm-<version>/lgtm/lgtm-config-gen.jar init
   ```

   You will now have a file called `lgtm-cluster-config.yml` in the `state` directory you created earlier.

2. If you have an SSL certificate—which is highly recommended for a production deployment (see "SSL certificates" on page 10)—open the newly created cluster configuration file in an editor and find the `ssl` block.

   Within this block, add entries for `certificate_path` and key_path, specifying the names of the files.

   ```
   ssl:
     hsts: false
     certificate_path: <certificate file>
     key_path: <key file>
   ```

   Note
   The paths to these files are relative to the cluster configuration file. If you have followed the instructions given here, this file is located in the `lgtm-releases/state` directory. This is the same place that you saved the certificate file and the key file,

so you can specify their location by just giving the file names, without a path.

Deploying a version of LGTM Enterprise

The following steps cover an initial installation, a redeployment, or an upgrade to a new version of LGTM Enterprise. A bash script containing these steps is available, as a worked example, on GitHub.com under the Semmle organization.

If you have followed the previous sections of this guide, the machine on which you are deploying LGTM already contains the following, within a directory such as `$HOME/lgtm-releases`:

```
.
├── lgtm-<version>
├── lgtm-<version>.tar.gz
└── state
    ├── lgtm-cluster-config.yml
    └── manifest.xml
```

1. Run the following command to generate the necessary files for your configuration of LGTM Enterprise:

   ```
   LGTM_CREDENTIALS_PASSWORD=<manifest-password> java -jar lgtm-<version>/lgtm/lgtm-config-gen.jar generate --input state/lgtm-cluster-config.yml --output lgtm-<version>/generated --overwrite
   ```

   The files are generated in the location specified by the `--output` value. These include a script called `install-machine.sh` which you will use shortly.

   > **Important**
   > The manifest password defined here is used to protect secrets normally used for inter-machine cluster communication. If you are performing an initial installation, make sure to retain this password as you will need it when you upgrade LGTM, or when you deploy changes you've made to the cluster configuration.

2. If LGTM is already installed, stop it by running:

   ```
   sudo lgtm-down
   ```

3. The `lgtm-config-gen.jar` tool created various files in the `lgtm-<version>/generated`

directory. Included in these files is a script called `install-machine.sh`. Run this now to install the necessary packages for LGTM Enterprise and to copy the files that were generated for your configuration of LGTM into the correct place on the local machine:

```
LGTM_DONT_START=true DEBIAN_FRONTEND=noninteractive sudo --preserve-env lgtm-
<version>/generated/localhost/install-machine.sh
```

> **Note**
> The `DEBIAN_FRONTEND` environment variable is only required for Debian and Ubuntu. It has no effect in other Linux distributions, so can be left in or omitted.

4. Start the core services that LGTM depends on, so that the database is available:

```
sudo lgtm-up --core-only
```

> **Note**
> *For a pre-1.20 upgrade only:* If you are upgrading to a version before 1.20.0 (for example, from 1.18.*x* to 1.19.*x*), replace the sudo `lgtm-up --core-only` command with these three commands:
>
> ```
> sudo systemctl start postgres-lgtm
> ```
>
> ```
> sudo systemctl start minio
> ```
>
> ```
> sudo systemctl start rabbitmq-server-lgtm
> ```

5. Run the following sequence of commands to create the database schema and perform any necessary upgrades:

> **Note**
> This sequence of commands is appropriate for any type of deployment (an installation, an upgrade, or a deployment of cluster changes). In all cases, run *all* of the following commands.

a. Create the schema (if necessary):

```
sudo lgtm-upgrade --action CREATE --if-not-exists --config
/etc/lgtm/config.json
```

> **Note**
>
> `config.json` is one of the files generated for your LGTM configuration when you run `lgtm-config-gen.jar`. It is copied to the `/etc/lgtm` directory when you run the `install-machine.sh` script.

b. Load the configuration for communication with other LGTM services

```
sudo lgtm-upgrade --action CONFIGURE --config /etc/lgtm/config.json
```

c. Ensure that an existing database has an up-to-date schema:

```
sudo lgtm-upgrade --action FULL --schema-only
```

d. Initialize the system and load the Semmle QL analysis engine:

```
sudo lgtm-upgrade --action INITIALIZE --core lgtm-<version>/lgtm/odasa-*.zip
```

e. Validate the database schema:

```
sudo lgtm-upgrade --action VALIDATE
```

f. Run any necessary data migrations:

```
sudo lgtm-upgrade --action FULL
```

> **Important**
>
> For an upgrade, the data migrations carried out by the `FULL` action of the `lgtm-upgrade` tool may take time proportional to the size of your instance of LGTM. So, for a very large instance of LGTM, this may cause an undesirable amount of downtime. You can, in this situation, start LGTM before carrying out the data migrations. To do this, run sudo `lgtm-up` immediately prior to running the `FULL` action.
>
> However, it's very important that these migrations are performed as you won't be able to upgrade a future version of LGTM Enterprise (other than a maintenance release) without having run them.

g. Check that the database is now fully up to date:

```
sudo lgtm-upgrade --action CHECK
```

6. Rename one of the files that was generated for your configuration, as follows:

   ```
   mv /etc/lgtm/config.json /etc/lgtm/config.initial.json
   ```

   Note

   The `config.json` file is never reused. Changing its name:

   a. Preserves a copy of the initial system configuration for reference.

   b. Prevents you from accidentally overwriting a current system configuration (which may contain changes made in the administrator interface) with the initial configuration, if you were to run the `lgtm-upgrade` tool's `CONFIGURE` action at a later date using the original JSON file.

Programmatically adding providers

Optional

Note

Generally this step only applies when you are installing LGTM Enterprise, or where you have tested integrations with third-party authentication, authorization, or repository providers on a test instance of LGTM and you want to deploy this configuration to a production instance.

If you have previously created a providers file (see "Including providers in a programmatic installation" on page 45), you can use this now to add preconfigured authentication, authorization, and repository providers to the new instance of LGTM Enterprise.

Run this command:

```
sudo lgtm-cli patch-config <path-to-providers-file>
```

Important

The providers file contains secrets—for example, the passwords of accounts on third-party systems—so you should take appropriate care to protect this file.

Start LGTM

Run this command:

```
sudo lgtm-up
```

Install the license

Before using a fresh installation, install the LGTM license that you uploaded to the LGTM host machine as one of the provisioning tasks:

```
sudo lgtm-cli license --install <path-to-license-file>
```

> **Note**
> For an upgrade, you can install a new license from the Settings page of LGTM's administrator interface.

Place reusable files under version control

Optional, but recommended

If you have not already done so, we recommend that you version control the `state` directory.

The `state` directory contains files that describe your LGTM instance. You will use these files again when you upgrade, or if you want to install another instance of LGTM Enterprise with a similar configuration. Placing this directory under version control means that you can simply check it out onto the appropriate server in future when you do an installation or upgrade.

Create a user account and log in

For installation only

Create an administrator account:

1. `sudo lgtm-cli create-user --email admin@example.com --password <password> --display-name Administrator`

2. `sudo lgtm-cli grant-admin --name admin@example.com`

You can now browse to your new LGTM Enterprise instance, log in using the account you just created, and start configuring LGTM Enterprise—see "Accessing the administrator interface" on page 48.

Distributed cluster programmatic deployment

Introduction

This section describes how to set up programmatic installation or upgrade for an LGTM Enterprise cluster that is distributed between multiple machines.

For details of the various components of an LGTM Enterprise cluster, see the LGTM Enterprise System Architecture PDF, or the "Server overview," "Control pool," and "Work pool" topics in the administrator help.

For the purposes of explaining how this can be done we're going to assume the following cluster topology, where the system is distributed between five virtual machines:

- 1 coordinator VM

  Hosting these cluster components:

  - Coordinator
  - Database
  - File store
  - Message queue
  - Search service

- 2 web host VMs*
- 2 worker host VMs (each running 3 workers)

> * Multiple web hosts assumes that the URL used to access LGTM Enterprise is served by two load-balanced web servers. The means by which you implement load balancing is beyond the scope of this documentation.

The system could be further distributed by putting some or all of the cluster components on separate machines. However, this is not usually necessary.

Resources on GitHub

If you are using Ansible to set up programmatic installation or upgrade, you may find it useful—as an alternative to creating a playbook from scratch—to download the following example of an Ansible playbook:

https://github.com/Semmle/lgtm-ansible-example

The instructions in the remainder of this section explain the commands used in a programmatic deployment and will help you to create the stored procedure you need for carrying out a programmatic deployment in any configuration management system.

Alternatively, if you prefer not to use a configuration management system, you may find it useful to download the following example bash script and accompanying the README file:

https://github.com/Semmle/lgtm-bash-deployment/tree/master/multi-machine

Note

The `deploy-multi.sh` file is provided purely as an example of a script that you could use to perform a distributed deployment.

For details of how to use this script to programmatically install or upgrade a distributed LGTM cluster see the README file that accompanies the script on GitHub.

Prerequisites

Your configuration management system playbook (or similar command sequence) should already include commands to create and provision the required virtual machines—as detailed on pages 25—27.

Before proceeding, make sure:

- *For installation only:* The "Prerequisites for installation" on page 9 have all been met.
- *For upgrade only:* You have read the relevant notes for the upgrade you're performing—see "Appendix—Notes for upgrading LGTM Enterprise" on page 51.

You can now continue with the main deployment stages we outlined earlier.

Design and generate the cluster configuration

In this stage of the process you will:

- Create a cluster configuration file (if you are installing and don't already have one)
- Use the cluster configuration file to generate the files you need to deploy the system

Create the cluster configuration file

Installation only

The cluster topology of an LGTM Enterprise instance is defined in the cluster configuration file: `lgtm-cluster-config.yml`. Your first task is to create this file.

Important

Given that the cluster configuration file is an input to the programmatic installation process it should be stored under version control so that it can easily be reused. Typically, the `state` directory that you created when provisioning machines for LGTM is version controlled.

Create a file called `lgtm-cluster-config.yml` in the `state` directory (for example, `$HOME/lgtm-releases/state`) with the following contents, replacing the variables highlighted in the example with actual values.

Alternatively, to create multiple clusters with the same topology, you could create a templated version of `lgtm-cluster-config.yml` (saving the template file under version control) and use commands stored in your configuration management system to produce the actual `lgtm-cluster-config.yml` file, inserting appropriate hostnames for the different cluster components. This would streamline the process of deploying multiple clusters programmatically (for example, a development cluster, a UAT cluster, and a production cluster).

Note

- You can download, or copy and paste, the following file from https://help.semmle.com/lgtm-enterprise/admin/help/installation/scripts/lgtm-cluster-config.yml.TEMPLATE. Copying and pasting directly from this PDF is not recommended as this may change characters and spacing.

- The variables shown below as *`<...-hostname>`* should all be replaced by the hostname by which the machine can be accessed by other machines in the cluster. The variable *`<LGTM-website-URL>`* must include a URL scheme and a hostname (or IP address)—for example, `https://lgtm.internal.example.com`. This is the URL at which users will access LGTM Enterprise.

- Environment variable expansion is not supported in the cluster configuration file.

```
version: 5
coordinator:
  hostname: "<coordinator-hostname>"
database:
  hostname: "<coordinator-hostname>"
file_storage:
  hostname: "<coordinator-hostname>"
queue:
  hostname: "<coordinator-hostname>"
search:
  hostname: "<coordinator-hostname>"
web:
  hosts:
  - hostname: "<webserver-1-hostname>"
  - hostname: "<webserver-2-hostname>"
  external_url: "<LGTM-website-URL>"
  ssl:
    certificate_path: "lgtm-web.crt"
    key_path: "lgtm-web.key"
workers:
  hosts:
  - hostname: "<workerhost-1-hostname>"
    general: 1
    query: 1
    on_demand: 1
    labels: []
  - hostname: "<workerhost-2-hostname>"
    general: 3
    labels: []
  temp_path: "/var/cache/lgtm/workers/"
data_path: "/var/lib/lgtm/"
```

Important

- The entries under `ssl:` in the configuration shown above assume that you have already generated SSL certificate and key files for the domain of the URL at which users will access LGTM, and you have saved these files on the coordinator host machine at the location suggested in the preliminary task "Locate the certificate, license, and providers files" on page 26.

  If required, you can use a self-signed certificate that LGTM generates, rather than using your own. To do this, remove the entries under `ssl:` in the configuration file. This is not recommended.

  For more information, see "SSL certificates" on page 10.

- Make sure that the lines are indented as shown above.

  You can check that the contents of the file are valid YAML by using a validation tool, such as `yamllint`. For more information, see yamllint.readthedocs.io.

Generate the LGTM configuration

The following series of commands use the cluster configuration file to generate the files that are required to install or upgrade the LGTM Enterprise cluster that the configuration file defines.

In your configuration management system playbook (or similar command sequence) add the following commands:

1. On the coordinator machine, check out the `state` directory from version control.

2. Run the following command on the coordinator machine. This command provides the password used to encrypt and decrypt information in LGTM's manifest file, and generates the files for your configuration of LGTM Enterprise:

   ```
   LGTM_CREDENTIALS_PASSWORD=<manifest-password> \

     java -jar <version-directory>/lgtm/lgtm-config-gen.jar \

       generate \

       --input <releases-directory>/state/lgtm-cluster-config.yml \

       --output <version-directory>/generated \

       --overwrite
   ```

   For example:

   ```
   LGTM_CREDENTIALS_PASSWORD=<password> \

     java -jar $HOME/lgtm-releases/lgtm-<version>/lgtm/lgtm-config-gen.jar \

       generate \

       --input $HOME/lgtm-releases/state/lgtm-cluster-config.yml \

       --output $HOME/lgtm-releases/lgtm-<version>/generated \

       --overwrite
   ```

   Important
   If you are copying the above command from this PDF, make sure that the hyphens shown above are retained when you paste the copied text.

   Note
   The password supplied in this command is used to encrypt and decrypt information in LGTM's manifest file.

> If you are installing LGTM Enterprise, make sure you retain the manifest password in a secure location. You will need it again when you upgrade LGTM, or when you deploy changes you've made to the cluster configuration file.
>
> You may want to save the password to a vault and include commands in your Ansible playbook (or similar) to extract the password for use in this step of the installation process.

Optional: snapshot the VMs

Upgrade only

Before proceeding with the upgrade, this is a good point to take a snapshot of each of the VMs in your LGTM cluster. This will allow you to redo the upgrade from this point if any of the commands you run afterward contain errors.

Install LGTM files on each machine in the cluster

The next main deployment stage (for upgrading as well as for installation) is to copy the version-specific release directory to each of the other machines in the cluster, and run a script to install the required software on that machine.

Add commands to your configuration management system playbook (or similar command sequence) to iterate through *each non-coordinator machine in the cluster* (for example, the worker host and web host machines) carrying out each of the following steps:

1. Copy the `lgtm-`*`<version>`* directory from the coordinator host machine to one of the other machines in the cluster (for example, a web application host machine or a worker host machine).

   > **Note**
   > The subdirectories of `lgtm-`*`<version>`*`/generated` that have the hostnames of machines other than the machine you are copying to can be omitted. Likewise you can omit the large `odasa-*.zip` file within the `lgtm-`*`<version>`*`/lgtm` directory.

   For example, after adding a private key for the current user on the local machine (the coordinator) and the corresponding public key on the remote machine, you could use rsync to copy the directory. The following rsync command copies the `lgtm-`*`<version>`* directory to a remote machine, excluding the `odasa-*.zip` file, and excluding the directories that relate to other cluster host machines. This ensures that unneeded files are not copied to the remote machine.

```
rsync --recursive --archive --compress --progress --include
"generated/<remote-machine-hostname>/" --exclude "generated/*/" --exclude
"lgtm/odasa-*.zip" $HOME/lgtm-releases/lgtm-<version>/ <user>@<remote-machine-
hostname>:$HOME/lgtm-releases/lgtm-<version>
```

2. *For upgrade only:* Stop LGTM by running the following command:

   - Linux machines:

     ```
     sudo lgtm-down
     ```

   - Windows worker host machines (as an administrator):

     ```
     %PROGRAMDATA%\LGTM\lgtm-worker\bin\lgtm-down.bat
     ```

   > **Note**
   > The order in which you bring down LGTM on the various machines is not significant.

3. *Linux only:* Set the `LGTM_DONT_START` environment variable to `true`, to avoid LGTM Enterprise attempting to start up when you run the next command:

   ```
   export LGTM_DONT_START=true
   ```

4. The `lgtm-<version>/generated` directory contains a subdirectory with the name of the current host. We now need to run an installation script located within this host-specific directory.

   Do the following on each non-coordinator machine in the cluster.

   - For Linux machines:

     ```
     sudo -E <version-directory>/generated/<local-hostname>/install-machine.sh
     ```

   - For Windows worker host machines:

     a. Copy the `lgtm-worker_<version>.msi` file from the `$HOME/lgtm-releases/lgtm-<version>/lgtm` directory to the `$HOME/lgtm-releases/lgtm-<version>/generated/<hostname>` directory:

     ```
     cp <version-directory>/lgtm/lgtm-worker_*.msi <version-
     directory>/generated/<local-hostname>/
     ```

b. Retrieve the logon details (domain\username and password) for the user that will be used to run worker daemons on this machine—for example, extract these from a vault in preparation for the next step.

> **Important**
>
> This user account must have full access to the working directory specified for this machine in the cluster configuration (`temp_path` property). The account details are passed directly to `sc.exe` and used to set up services for LGTM.

c. Run the `install-machine.bat` script as the user account whose logon details you retrieved, using the syntax:

```
install-machine.bat <domain>\<username><logon-password>
```

For example:

```
<version-directory>/generated/<local-hostname>/install-machine.bat
.\lgtmuser <lgtmuser-password>
```

Once this is done, do the following on the coordinator machine:

1. *For upgrading only:* `sudo lgtm-down`

2. `export LGTM_DONT_START=true`

3. `sudo -E <version-directory>/generated/<local-hostname>/install-machine.sh`

Initialize the LGTM database and start all components

The final stage of the deployment involves starting the database, message queue, and file storage components, initializing or upgrading the database, and then bringing up the system.

Add commands to your configuration management system playbook (or similar command sequence) to perform the following actions on the appropriate machines:

1. Bring up the core services for LGTM by running:

   ```
   sudo lgtm-up --core-only
   ```

   Run this command on machines that host any of these cluster components:

   - LGTM database

- File storage service
- Message queue

In our example, these components are all hosted on the same machine as the coordinator, so this command only needs to be run on the coordinator machine.

> **Note**
> *For a pre-1.20 upgrade only:* If you are upgrading to a version before 1.20.0 (for example, from 1.18.*x* to 1.19.*x*), replace the sudo `lgtm-up --core-only` command with these three commands:
>
> `sudo systemctl start postgres-lgtm`
>
> `sudo systemctl start minio`
>
> `sudo systemctl start rabbitmq-server-lgtm`

2. On the coordinator machine, initialize or upgrade the LGTM database by running the following commands.

> **Note**
> Run the following sequence of commands whether you are performing an installation or an upgrade.

a.
```
sudo lgtm-upgrade --action CREATE \
  --if-not-exists --config /etc/lgtm/config.json
```

> **Note**
> `config.json` is one of the files generated for your LGTM configuration when you run `lgtm-config-gen.jar`. It is copied to the `/etc/lgtm` directory when you run the `install-machine.sh` script.

b.
```
sudo lgtm-upgrade --action CONFIGURE \
  --config /etc/lgtm/config.json
```

c. `sudo lgtm-upgrade --action FULL --schema-only`

d. `sudo lgtm-upgrade --action INITIALIZE \`

```
  --core <version-directory>/lgtm/odasa-*.zip
```

e. `sudo lgtm-upgrade --action VALIDATE`

f. `sudo lgtm-upgrade --action FULL`

g. `sudo lgtm-upgrade --action CHECK`

3. On the coordinator machine, rename the system configuration file as follows:

   ```
   mv /etc/lgtm/config.json /etc/lgtm/config.initial.json
   ```

   Note
   The `config.json` file is never reused. Changing its name:

   a. Preserves a copy of the initial system configuration for reference.

   b. Prevents you from accidentally overwriting a current system configuration (which may contain changes made in the administrator interface) with the initial configuration, if you were to run the `lgtm-upgrade` tool's `CONFIGURE` action at a later date using the original JSON file.

4. On each machine in the cluster, bring up LGTM Enterprise by running the following command :

   - Linux machines:

     ```
     sudo lgtm-up
     ```

   - Windows worker host machines (as an administrator):

     ```
     %PROGRAMDATA%\LGTM\lgtm-worker\bin\lgtm-up.bat
     ```

> **Note**
> The order in which you bring up LGTM on the various machines is not significant.

5. *For installation only:* On the coordinator machine, create a user account that you will use to log on to LGTM Enterprise and grant the account administrator privileges so that it has access to the administrator interface:

   ```
   sudo lgtm-cli create-user --email <admin-user-email-address> \

     --password <admin-user-email-address> \

     --display-name Administrator

   sudo lgtm-cli grant-admin --name <admin-user-email-address>
   ```

   > **Note**
   > The user can change the password specified here (their email address) and the display name ("Administrator") after logging in.

6. On the coordinator machine, install your LGTM Enterprise license:

   ```
   sudo lgtm-cli license --install <version-directory>/lgtm/license.dat
   ```

7. *Optional—for installation only:* If you have previously created a providers file (see "Including providers in a programmatic installation" on the next page), use this now to add preconfigured authentication, authorization, and repository providers to the new instance of LGTM Enterprise.

   On the coordinator machine, run:

   ```
   sudo lgtm-cli patch-config <path-to-providers-file>
   ```

   > **Important**
   > The providers file contains secrets—for example, the passwords of accounts on third-party systems—so you should take appropriate care to protect this file.

If you are performing an installation, you can now go to the administration interface and start configuring LGTM Enterprise—see "Accessing the administrator interface" on page 48.

If you are upgrading LGTM Enterprise, check the relevant After you upgrade to... section of the appendix, starting on page 51, for any additional steps you need to take to configure the new version.

Including providers in a programmatic installation

LGTM "providers" provide the integration between LGTM and third-party authentication, authorization, and repository systems. For more information about LGTM providers, see the topic on "Defining integrations" in the administrator help.

Typically, you will want a programmatic installation to add providers to the resulting system, so that you don't have to add them manually after installation, each time you install LGTM Enterprise.

Before you can include providers in a programmatic installation, you need to have first configured the providers in the administration interface of an existing instance of LGTM Enterprise.

Including providers in the commands for a programmatic installation therefore involves:

1. Installing LGTM Enterprise—for example, using the interactive installation method.
2. Adding authentication, authorization and repository providers as required (instructions are provided in the administrator help).
3. Saving the provider settings as a JSON file (see below).
4. Passing the providers JSON file to the `lgtm-cli patch-config` command at the end of a programmatic installation.

Saving the provider settings

Assuming you have configured providers in a existing LGTM Enterprise instance, you can capture the settings as follows:

1. Working in a console window, on the LGTM coordinator machine, change to your user's home directory.

   Tip
   If the command-line JSON processor `jq` is installed on the host machine, you can use the following commands as a shortcut, instead of steps 2—7 below. Run these commands and then skip to step 8:

   ```
   sudo lgtm-cli get-config > running-config.json
   jq "{ access: .access }" running-config.json > providers.json
   rm running-config.json
   ```

2. Use the `get-config` action of the `lgtm-cli` command to output the current running

configuration to a file:

```
sudo lgtm-cli get-config > providers.json
```

This creates a file called `providers.json` in the current directory containing details of the running configuration.

> ### Terminology
>
> **Running configuration**: data within LGTM's internal database that describes the currently running system. Not to be confused with the cluster configuration file, which defines the topology of an LGTM Enterprise cluster.

3. Open the file for editing.

   It contains JSON-formatted data, enclosed in braces: `{ ... }`

4. After the initial opening brace, delete everything up to the line:

   ```
   "access" : {
   ```

5. Find the matching closing brace for this section.

6. Delete everything between this closing brace and the final closing brace in the file, making sure to delete the comma after the second last closing brace in the file.

   The file should now comprise:

   ```
   {
       "access" : {
           <details of various types of providers>
       }
   }
   ```

   The remaining JSON describes the providers you have set up—including the authentication details used to access those services.

   You can use this file to load these providers into LGTM Enterprise at the end of the programmatic installation process.

7. *Optional:* Check that the file is valid JSON by using a tool such as `jq`.

Running `jq . <path-to-file>` will display the file's contents if it's valid JSON, and if not will display information about errors in the file. For more information, see stedolan.github.io/jq/manual.

8. Move the `providers.json` file to a secure location for use when you perform a programmatic installation.

> **Important**
> The providers file contains secrets—for example, the passwords of accounts on third-party systems—so you should take appropriate care to protect this file.

Adding the provider settings to the programmatic installation

Now that you have saved the provider settings to a file you can use the `lgtm-cli patch-config` command to load these settings, as the final step in a programmatic installation (see the instructions on page 32 for a single-machine programmatic deployment, or page 44 for a distributed cluster programmatic deployment).

Post-installation procedures

After installing LGTM Enterprise you can log into the application, go to the administration pages, perform some basic configuration, and test the system.

Accessing the administrator interface

1. In a browser, go to https://*<LGTM-domain>*/admin.

   The LGTM Enterprise log in page is displayed.

2. Log in to LGTM Enterprise using the administrator account credentials used during installation.

3. If prompted to do so, choose a user name, or accept the default, and click **Let's go**.

   The LGTM Enterprise administration home page is displayed.

4. To see information about configuring and administering LGTM Enterprise, click **Admin help** in the menu bar. This displays LGTM's built-in administrator help.

> **Tip**
> A searchable version of the administrator help is available at help.semmle.com.

Useful help topics

The following topics on help.semmle.com (also available within the application's administrator help) provide useful information on administrative tasks you might want to carry out:

- Change the administrator user's password from the default assigned during installation
- Configuring the environment on the worker host machines to allow workers to analyze code
- Defining "providers" to allow LGTM to access source code repositories, and to allow users appropriate access to project results
- Adding projects for initial analysis

- Removing projects added for testing
- Checking the progress of analysis for projects you have recently added
- Assessing worker resourcing: do you have too few, or too many, worker daemons?
- Adding and removing worker daemons
- Editing and deploying the cluster configuration—for example, to add more worker host machines
- Creating user accounts
- Viewing analysis results as a standard LGTM user
- Making LGTM's plugins available for people to install in their software development applications

Post-upgrade procedures

Check the relevant After you upgrade to... section in the appendix, starting on page 51, for any additional steps you need to take to configure the new version.

Post-upgrade housekeeping

After upgrading LGTM Enterprise, if you have used the directory structure suggested in this documentation, you'll have the following directories:

- `<path-to>/lgtm-releases/lgtm-<old-version>`
- `<path-to>/lgtm-releases/lgtm-<current-version>`

You can delete the `lgtm-<old-version>` directory. It's not used after you've upgraded. For example, you might delete the `lgtm-1.19.3` directory after upgrading to LGTM Enterprise 1.20.

You can also delete the `lgtm-<version>.tar.gz` distribution file.

Appendix—Notes for upgrading LGTM Enterprise

Important
You must upgrade to each version in turn. This ensures that any data migration required by each upgrade is completed successfully before you begin the upgrade to the next version. This does not include maintenance releases so, for example, you can upgrade from any 1.19.x version to any 1.20.x version.

If you have made any manual changes to the configuration of LGTM, beyond editing the cluster configuration file, upgrading may overwrite these changes. We recommend that you back up your system before upgrading.

Before proceeding with an upgrade, read the relevant section below.

You can then proceed, and perform either an interactive upgrade or a programmatic upgrade.

Upgrading from 1.19 to 1.20

After you upgrade to 1.20

Contact Semmle if you have been running workers on Windows—Version 1.20 of LGTM Enterprise introduces a new, streamlined system of installing, upgrading and maintaining worker daemons on Windows. If you're already running workers on Windows, you can continue using these after upgrading. However, you should talk to your Semmle support contact about migrating these workers to the new system to ensure compatibility with future updates.

Note
If you have not yet used Windows machines to run worker daemons and you now wish to do so, there is no need to contact Semmle. Instructions for adding Windows worker hosts is now included the administrator help—see Adding new hosts to the work pool.

Upgrading from 1.18 to 1.19

After you upgrade to 1.19

Caution

All repository providers, other than those for Subversion and simple Git, must now specify a user name and password. This was strongly recommended in version 1.18 of LGTM Enterprise, but is now mandatory in 1.19. The reasons behind this change are described in security advisory SSA-2018-004.

After upgrading you must ensure that the settings for all repository providers (except Subversion and simple Git) include the name and credentials for a valid account. Check the Administration interface page for the error message: "No new commits will be analyzed for projects hosted on repository provider <*name*>. Please add credentials to its configuration page."

The account is used by LGTM Enterprise:

- To determine a variety of metadata about repositories—for example, whether the repository name has changed
- To post comments and set statuses on the pull requests of projects for which automated code review integration has been set up

For more information about repository providers, see Adding repository providers.

Enabling custom queries—To enable users to add custom queries to your LGTM instance, make sure to set an organization identifier on the Settings page of the administrator interface. See Defining your organization identifier.

Upgrading from 1.17 to 1.18

After you upgrade to 1.18

After upgrading to 1.18, you should make the following changes:

- **Define accounts for repository providers**—this release adds user account fields to the configuration forms for all external repository providers. Specifying user account details is optional in this release but is strongly advised, and is mandatory in 1.19. Doing so ensures that LGTM will analyze the correct code for a project following a rename or a change to the clone protocol or default branch. For more information, see Adding external repository providers.

- **Update links to LGTM Enterprise logs**—the URL for job logs has changed in this release. A `jobs` directory has been added to the URL. Any saved links containing paths such as `/admin/logs/3504840010` should be updated to add `jobs` after the `logs` directory—for example, `https://<LGTM-server>/admin/logs/jobs/3504840010/stdout`. A URL redirection in this release ensures that links that use the old path will still work for now. However, this redirection will be removed at the next release so it's a good idea to advise users to update any links they have saved now.

- **Make the 1.18 IDE plugins available to users**—from 1.17.1 onward, the LGTM IDE plugins use a new method to authorize access to LGTM Enterprise. Users can only download alerts from LGTM 1.18 if they upgrade to at least version 1.17.1 of the plugin.

Upgrading from 1.16 to 1.17

Before you upgrade to 1.17

Before you upgrade from 1.16.x to 1.17.x, you should consider the following changes:

- **Export of business intelligence (BI) data will stop**—there is a new database format for 1.17.

> **Caution**
> If your system is set to export LGTM data to a database, this is automatically disabled by upgrading to 1.17. After upgrading you need to prepare a database before you restart export. For detailed information, see "Re-exporting BI data from scratch" in the administrator's help.

- **Dedicated search service is required**—this was introduced in 1.16. If you originally installed 1.15 and have not enabled this service manually, you'll be prompted to do this as part of the upgrade.

After you upgrade to 1.17

After upgrading to 1.17, you may want to make the following changes:

- **Configure an empty database and re-enable BI export**—if you export data from LGTM for use with BI tools, you need to make an empty database available for the new data and set up

BI export in the administration pages of the application. For more information, see Re-exporting BI data from scratch.

- **Add on-demand workers**—consider updating the cluster configuration to include some of the new on_demand workers. For more information, see Worker types.
- **Configure workers for Team Foundation Version Control (TFVC) analysis**—if your organization stores code in TFVC repositories, you may want to configure some or all of the worker hosts to checkout code from these repositories. For more information, see Installing a command-line client for TFVC.

Upgrading from 1.15 to 1.16

Before you upgrade to 1.16

If you haven't upgrade to 1.16.x yet, you should also consider the following changes:

- **Different ports required**—workers now download jobs from the job dealer service via **port 8443**. Previously they accessed the message queue (an instance of RabbitMQ) in the control pool directly via port 5671. Also, the web application communicates with the new search component via TCP on **port 8983**. You need to review your firewall and network configurations and ensure that these communication paths are available. For the full architecture diagram, see Network connections.
- **New format for the cluster configuration file**—your cluster configuration file is updated automatically to the new format during the upgrade process.

> **Note**
> If your system uses an SSL certificate that wasn't generated by LGTM, you should back up the certificate and key files before you upgrade. The upgrade process from 1.15 to 1.16 will overwrite your certificate with an autogenerated certificate.

- **Additional worker host requirements**—JavaScript analysis is extended to include TypeScript (in `.ts` and `.tsx` files). If any analyzed project includes TypeScript files, JavaScript analysis will fail unless the additional requirements are available. For information about configuring the work pool for TypeScript analysis, see Setting up a host for JavaScript and TypeScript analysis.

After you upgrade to 1.16

After upgrading to 1.16, you should plan to:

- **Add your SSL certificate to the cluster configuration**—update the cluster configuration file to specify the location of the SSL certificate files to use. This will ensure that the correct SSL certificate is automatically used and maintained through future upgrades. For more information, see the revised topic on installing SSL certificates.
- **Make the IDE plugins available to users**—users can uses these to download alerts from LGTM Enterprise and display the alongside the code in their IDE. You can download the installers from the customer wiki and make them available to your users. For more information, see Making IDE plugins available to users.