

lgtm enterprise



LGTM Enterprise System Requirements

Release 1.22, August 2019

SemmleTM

Semmler Inc.

44 Montgomery Street
3rd Floor
San Francisco, CA 94104

Copyright © 2019, Semmler Ltd. All rights reserved.

LGTM Enterprise release 1.22

Document published August 27, 2019

Contents

Introduction	4
Languages supported	6
System requirements	7
Work pool software requirements	10
Deployment	13
System integrations	16
Browser support	18

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 document describes the minimum system requirements for a deployment of LGTM Enterprise. For more detailed information about the services mentioned in this document, see the [administrator help](#).

Related documentation

- [LGTM Enterprise System Architecture](#) (PDF)
- [LGTM Enterprise Installation and Upgrade Guide](#) (PDF)
- LGTM Enterprise administrator help

To access this, click **Admin help** at the top of the administration pages in LGTM Enterprise, or browse to help.semmle.com/lgtm-enterprise/admin.

Languages supported

LGTM Enterprise supports analysis of code written using the following languages:

- C and C++
- C#
- COBOL
- Java
- JavaScript/TypeScript
- Python

For detailed information about which versions of these languages are supported, and details of built-in support for frameworks, see [Languages and compilers](#).

System requirements

In some systems the control pool consists of a single machine—referred to as the "controller." This also hosts the web and data aggregation services so that there is no separate web pool. At larger scales a web pool of additional machines can be added to increase capacity for web and data aggregation services, with the "controller" remaining on a separate machine.

Type	Hardware	Software
Control pool*	Controller <ul style="list-style-type: none"> • 8 cores • 16 GB RAM • Solid-state drive (SSD)† • 1 TB free disk space 	Operating system —one of the following Linux distributions/versions: <ul style="list-style-type: none"> • CentOS 7 • Debian 9 • Red Hat Enterprise Linux 7.x (from 7.4) • Ubuntu 16.04 LTS • Ubuntu 18.04 LTS
	Web pool* <ul style="list-style-type: none"> • 2 cores • 8 GB RAM • SSD recommended† • 50 GB free disk space 	
Work pool*	<ul style="list-style-type: none"> • 2 cores • 8 GB RAM • SSD recommended† • 50 GB free disk space (per worker assigned) 	Operating system —Linux versions as above, or Windows 7 or later. Required on Windows: <ul style="list-style-type: none"> • The 64-bit version of the most recent release of OpenJDK 8, or Oracle JDK 8. Note: IBM Java, and other versions of Java are not supported. • Python 2.7 On Linux, the latest JDK 8 and Python 2.x are installed on the worker host machine by LGTM if they are not found.

* For more information, see the [LGTM Enterprise System Architecture](#) PDF, or the "[Server overview](#)" topic in the LGTM Enterprise administrator help.

† Semmler recommends that you use solid-state drives (SSDs) rather than hard disk drives (HDDs) for all of the machines, in both the control pool and the work pool. It is strongly

recommended that you use an SSD for the "controller" machine, as files are served from here to all other cluster machines.

Note

During installation, upgrade and configuration processes, root access is required for each machine.

Virtual machines (for example, using VMWare, Xen, or KVM) or physical machines are both suitable for the pools of machines. The terms "machines" and "virtual machines" are used interchangeably in the documentation.

The computational power required for the work pool depends on the number of projects to analyze, their size, build times (for compiled languages such as Java), and the average commit frequency.

As this information is usually difficult to ascertain in advance, it makes sense to plan for an incremental deployment based on the following information:

- Minimum deployment size for LGTM Enterprise
- Hardware requirements at scale
- Incrementally adding capacity

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 2019-05-02T19-07-09Z (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 tools 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.

Work pool software requirements

The LGTM work pool must be set up with machines that include the right software and the correct environment to enable checkout and analysis of the codebases you want to analyze.

Checkout requirements

The essential requirements are:

1. **Appropriate client software** to allow the worker to check out code from the repository— that is, Git, Subversion or Team Foundation Version Control (TFVC). The LGTM Enterprise installer automatically installs Git and Subversion clients. For TFVC, you must install the Team Explorer Everywhere command-line client (TEE-CLC) on each worker host machine.
2. **Repository access** for all worker daemons. All of the worker daemons running on machines in the work pool need to be able to connect to all of the repository hosts you use. They also need read access to the code.

Note

All general and on-demand workers can be used to poll your repositories, to check whether new changes have been made to the code. This is true even for workers that will not be used to build and analyze code from a particular repository. So, typically, connections to all of your repository hosts occur from all of your worker host machines.

Generally, workers use the authentication credentials you specify when you create a repository provider (see "[Adding repository providers](#)" in the administrator help) to access the repositories. For Git-based repository providers, however, you have the option of configuring workers to connect over SSH, by setting the provider's **Checkout method** accordingly.

For more information, see "[Configuring hosts to checkout code](#)" in the LGTM Enterprise administrator help.

Analysis requirements

During the analysis of compiled languages, LGTM builds the code. This means that there must be at least one machine in the work pool that is configured with the environment needed to

build each codebase that you want to analyze. Interpreted languages often also depend on the presence of other software in the environment, and these must be made available too.

Core requirements

The core requirements listed below are the minimum required to analyze code written in each language. Your projects may have additional requirements. Compiled languages have further requirements (detailed in the following section).

Language	Essential
C/C++	Method of building the code (see next table).
C#	Method of building the code (see next table).
COBOL	No further requirements.
Java	Method of building the code (see next table).
JavaScript	No further requirements unless any TypeScript files are present. If the repository also includes TypeScript files, the TypeScript requirements must be met, or TypeScript analysis disabled.
Python	Access to the pip packaging management system. Installation of the packaging and virtualenv pip modules. Ability to download any packages that the Python codebases you intend to analyze depend on.
TypeScript	Node.js version 6.x or later.

Compiled languages

For compiled code that is normally built using one of the most commonly used build systems (listed in the following table), LGTM Enterprise has an "autobuild" process which is designed to automatically build the code. If your organization uses these build systems, install them on the relevant worker machines. If some projects use other build systems, you also need to make these systems available. These projects will require a custom build configuration.

Language	Autobuild available for
C/C++	Linux workers: Autoconf, CMake, QMake, Meson, WAF, SCons, and Linux KBuild Windows workers: MSBuild
C#	MSBuild, .NET Core, Mono (Linux), and Visual Studio (Windows)
Java	Gradle, Maven and Ant

For more information, see "[Configuring hosts to analyze code](#)" in the administrator help.

Deployment

The software is deployed to Linux machines using packages. Semmler provides packages in common formats (RPMs and Debian packages) for the LGTM Enterprise software and for those dependencies that are not available by default through recent Linux distributions.

Machines in the work pool should be configured to checkout and build the software that they will analyze and tested before LGTM is deployed to them. LGTM is deployed using LGTM RPM or Debian packages on Linux machines, and an MSI installer on Windows machines.

Minimum deployment

For an initial trial, a suitable cluster of machines is:

- Control pool: 1 virtual machine (VM)
- Work pool: 5–20 virtual machines

The hardware requirements defined in the [system requirements](#) are enough to support one worker per work pool VM. If you want to run more than one general worker per work pool VM, multiply the disk space, RAM and CPU cores accordingly.

Note

When you add projects with over 1 million lines of code to LGTM, you may need to increase the resources available on the work pool machines. More RAM and disk space may be required to build large projects. If the builds are parallelized, increasing the number of cores available on each machine may also be beneficial.

In addition to the hardware and software requirements above, each VM in the work pool must be set up with the build environment that is required to build the projects that will be analyzed. For example:

- For C# analysis the correct versions of the C# compiler must be installed.
- For Python analysis pip and the packaging module must be installed. In addition, to analyze code for Python 2 projects, the virtualenv module must be installed.

Hardware requirements at scale

LGTM Enterprise is designed to scale successfully from analyzing the work of a few hundred developers to analyzing the work of many thousands of developers. The resources required

change according to the nature of the repositories analyzed, size of the projects, number of developers and commit frequency. We recommend that you start with a minimum deployment and scale this as required.

You can monitor the free capacity of the work pool using the Infrastructure administration page. If you need to add extra capacity, you should provision additional machines and register them as part of the work pool.

In rare cases where there is a particularly large amount of data, the services on the controller can be hosted on different machines.

Other bottlenecks are usually best addressed by changing the control pool from a single machine to a cluster. For example:

- Create a web pool of several machines to load balance the web application and web proxies. These machines also process the results returned by the work pool.
- Host the database on a dedicated virtual machine with substantial storage.
- Host the file store on a dedicated virtual machine with substantial storage.

Examples

There's an enormous variation in how companies organize their source code and development workflow. This makes it difficult to predict the hardware requirements for LGTM Enterprise in advance of installation, which is why LGTM is designed to scale smoothly. To help you get an idea of the likely requirements, here are some examples of the resources used by production instances of LGTM.

LGTM.com

This instance analyzes the work of over half a million developers working on code in over eighty thousand repositories. The following resources are used on Google Cloud Platform:

- A dynamic control pool that scales up and down automatically according to the current workload.
- A dynamic work pool that typically runs between 600 and 1250 workers. Each worker runs on a separate, [preemptible](#), virtual machine with 2 vCPUs, 7.5 GB of system memory, and 80 GB of disk space.

LGTM Enterprise

In this example, the instance of LGTM Enterprise analyzes the work of approximately 2,500 developers working on code in over 850 repositories. The following internal virtual machines are used:

- One controller running on a virtual machine with 6 vCPUs 2.6 GHz, 32 GB of system memory, and 1 TB disk space.

- A work pool containing 15 general workers and 1 query worker. These are spread across 8 virtual machines. Each of these virtual machines has 6 vCPUs 2.6 GHz, and 32 GB of system memory.

Incrementally adding and removing capacity

As new projects are added to LGTM Enterprise, the work pool (and occasionally the control pool) will need to be expanded. Machines in the work pool are stateless and can all be configured identically. This makes it easy to scale the pool up and down elastically in response to demand, by provisioning virtual machines from a common image.

System integrations

Integration	User authentication	User authorization	Code review integration
Azure DevOps Server (previously called TFS) 2017 or later (Git and TFVC projects)	✓		
Azure DevOps Services (previously called VSTS) (Git and TFVC projects)	✓	✓	Git projects only
Bitbucket Cloud (Git projects only)	✓	✓	✓
Bitbucket Server 4.5.0 or later (Git projects only)	✓	✓	✓ *
<i>Git—repository access only</i>			
GitHub.com	✓	✓	✓
GitHub Enterprise 2.9 or later	✓	✓	✓
GitLab.com	✓	✓	✓
GitLab Enterprise and Community Edition 9.0 or later	✓	✓	✓
OneLogin (SAML authentication)	✓		
<i>Subversion—repository access only</i> 1.8 or later			

* Requires that the Bitbucket Server trusts the LGTM instance's SSL certificate, and has the Post Webhooks plugin installed. For the latest information about which versions of Bitbucket the plugin supports, see the [plugin's documentation](#).

Private repositories

User authorization ensures that access rights to projects, configured in a host system, are respected in LGTM Enterprise.

LGTM Enterprise supports adding projects from private repositories hosted on any of the supported cloud-based services for which there is user authorization.

Issue tracker integration

You can configure LGTM Enterprise to create issues in external issue tracking systems for alerts found by LGTM. Integration is configured using webhooks—see the topic "[Adding webhook-based issue tracker integration](#)" in the administrator help.

For Atlassian Jira version 7.6.9 or later, the LGTM Jira add-on provides an easy way to generate issues from LGTM alerts. For more information, see the topic "[Adding Jira integration](#)" in the administrator help.

Browser support

The following web browsers are supported for accessing the web pages of an LGTM Enterprise deployment:

- Google Chrome
- Mozilla Firefox
- Apple Safari
- Microsoft Edge

Semmler aims to provide an optimal user experience for the most recent version of all of the browsers mentioned above, and will support browser versions for up to three years after their release.