lgtm enterprise

LGTM Enterprise System Architecture

Release 1.23, December 2019

Semmle™

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

LGTM Enterprise release 1.23
Document published December 16, 2019

Contents

Introduction

About this document

This document is an excerpt from the LGTM Enterprise administrator help. It provides a series of basic architecture diagrams intended to provide a high-level overview of the main components of LGTM Enterprise and how they are connected. For more detailed information about the services mentioned in this document, see the LGTM Enterprise administrator help.

Related documentation

- *LGTM Enterprise System Requirements* (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.

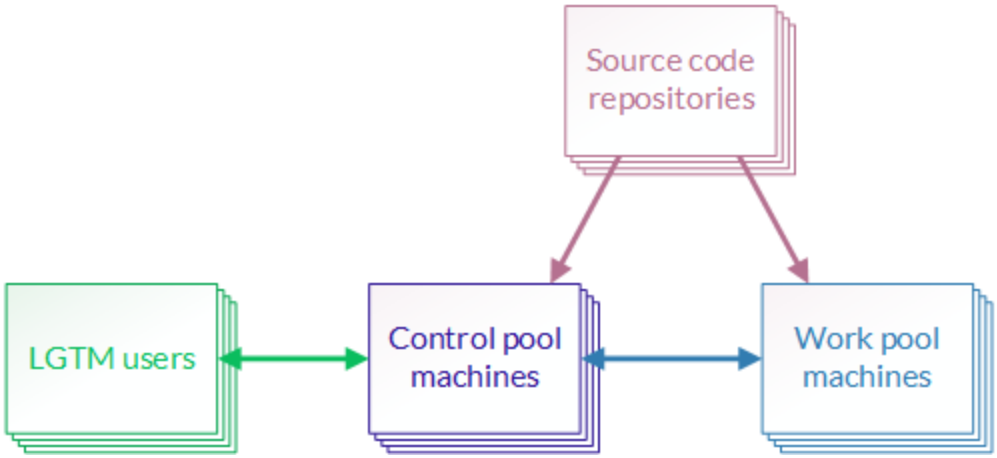
System architecture overview

The machines used in a deployment of LGTM Enterprise can be classified into two types:

- **Work pool** machines
  The machines in the work pool host one or more LGTM workers (processes that build code and generate analysis data). For more information, see "Work pool" on page 11.
- **Control pool** machines
  The machines in the control pool:
  - Store the persistent state of the LGTM cluster
  - Coordinate the work of the work pool, process and store its results
  - Host the web interface to the cluster

  For more information, see "Control pool" on page 9.

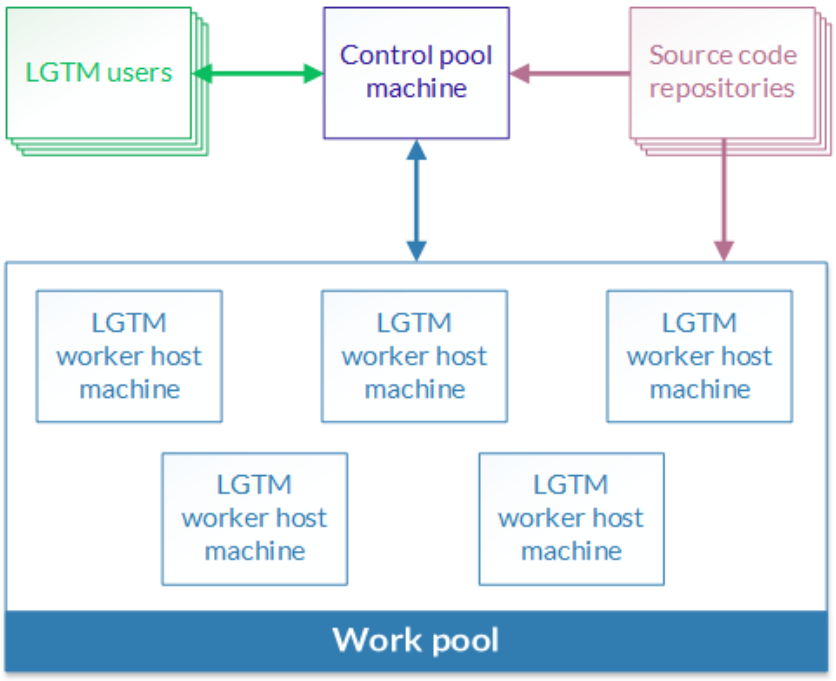
Overview of data flow:



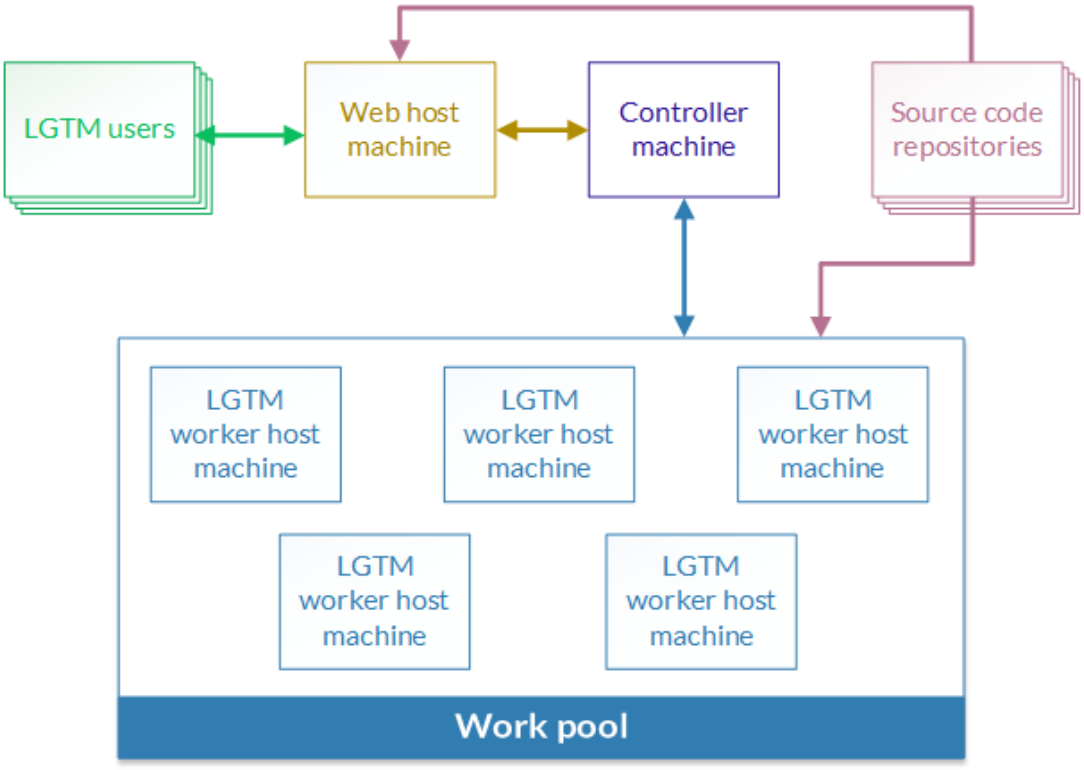
Note
Except where otherwise stated, the arrow directions on diagrams give an indication of data flow between system components.

Basic architecture

For a simple deployment of LGTM Enterprise you might have just one control pool machine and perhaps five work pool machines:



Alternatively, the web interface is commonly hosted on a separate machine from the other control pool services, which are all hosted on a "controller" machine:



The other services in the control pool can also be placed on separate machines. For more information, and details about the ports that must be externally reachable if services are placed on separate machines, see "Network connections" on page 13.

Cluster configuration

You configure the topology of your LGTM Enterprise instance in a cluster configuration file. You then deploy this configuration to the machines specified in the file.

The following extract from a cluster configuration file shows the topology described by the diagram above: one controller machine, one web host machine, and five worker host machines:

```
...
coordinator:
  hostname: "lgtm-controller"
database:
  hostname: "lgtm-controller"
file_storage:
  hostname: "lgtm-controller"
queue:
  hostname: "lgtm-controller"
search:
  hostname: "lgtm-controller"
task_workers:
  hosts:
  - hostname: "lgtm-controller"
web:
  hosts:
  - hostname: "lgtm-web"
  ...
workers:
  hosts:
  - hostname: "lgtm-workerhost1"
    ...
  - hostname: "lgtm-workerhost2"
    ...
  - hostname: "lgtm-workerhost3"
    ...
  - hostname: "lgtm-workerhost4"
    ...
  - hostname: "lgtm-workerhost5"
    ...
```

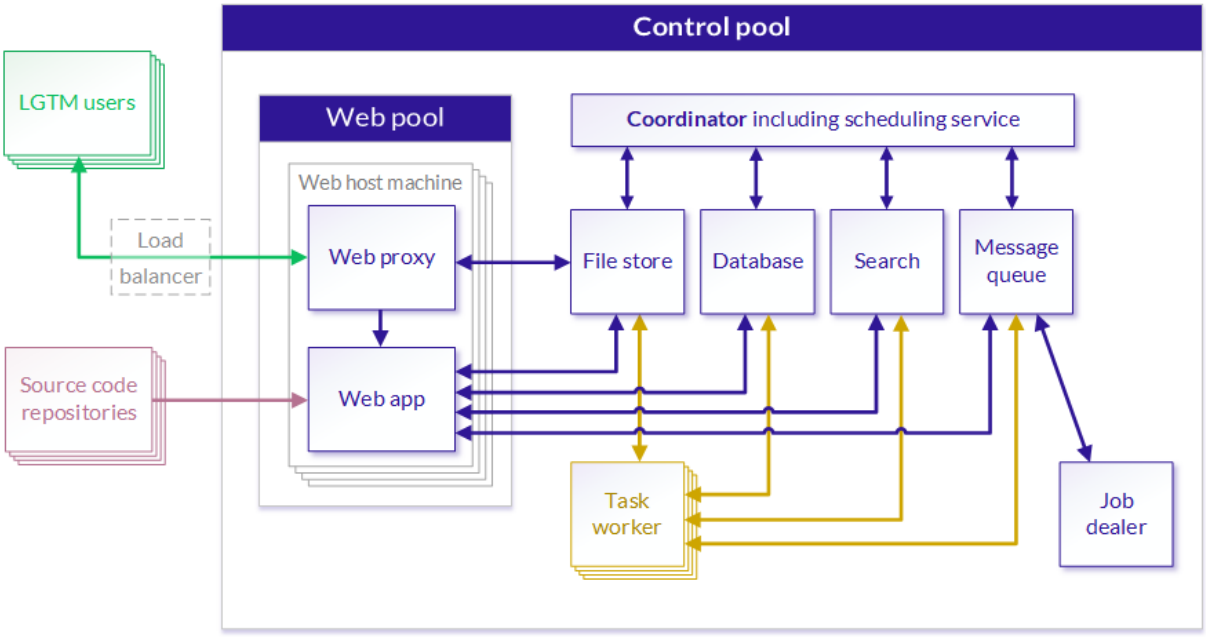
For full details of the cluster configuration file, see the topic "lgtm-cluster-config.yml file" in the LGTM Enterprise administrator help.

Control pool

The control pool is made up of one or more machines. These host:

- Scheduling service—runs periodic tasks at set intervals.
- Database service—provides permanent storage of LGTM data.
- File storage service—provides storage of frequently accessed data. This gives immediate access to repositories in LGTM format, recent build/analysis log files, unprocessed job results, and the latest CodeQL database for each project (used by the query console).
- Message queue service—manages work. The associated job dealer ensures that build/analysis jobs are processed by a suitable worker.
- Search service— provides help search functionality.
- Task worker service—loads the results of analysis jobs into the LGTM database and performs other background processing.
- Web pool which runs:
  - Web application—handles requests to the LGTM web interface.
  - Web proxy service—acts as a proxy and SSL terminator for the LGTM web interface.

For more information about these services, see the topic "Services" in the LGTM Enterprise administrator help.



The machine that hosts the scheduling service is referred to as the coordinator. If the control pool components are distributed across multiple machines, it is important to know which

machine is the coordinator because LGTM command-line actions must be run from this machine.

Work pool

The work pool consists of one or more "worker" host machines (or containers in a Dockerized deployment). The workers hosted on the machines in the work pool are daemons that carry out the "heavy lifting" tasks of building and analyzing code.

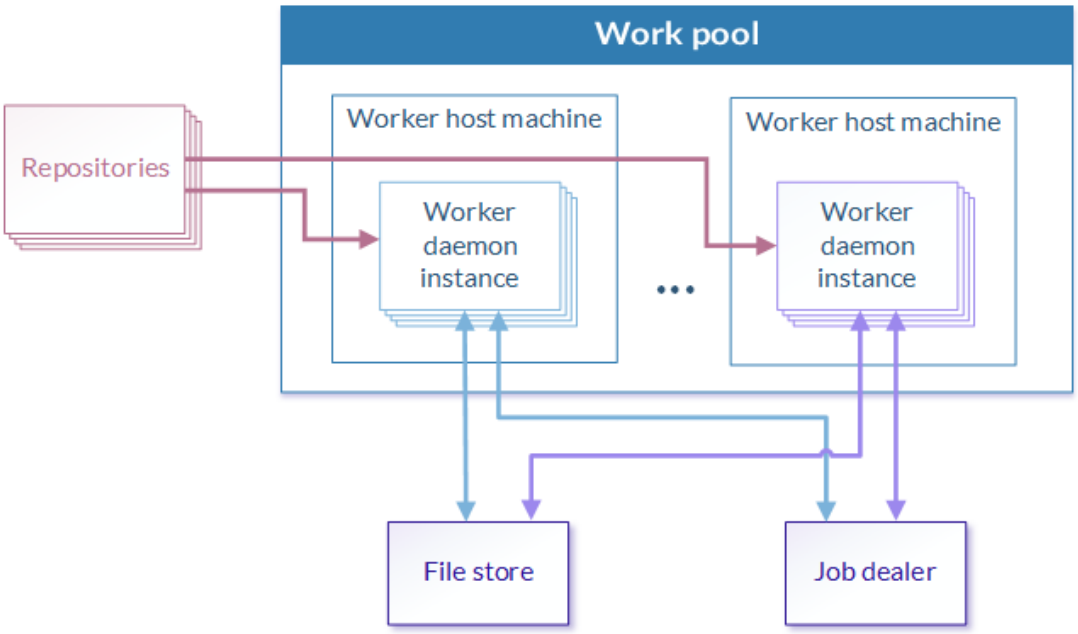
In the simplest, non-Dockerized, demo setup the work pool and the control pool may be hosted on the same machine and you may only run the minimum of two worker daemons (one for processing users' custom queries, one for all other work pool tasks).

For deployments, the work pool will consist of multiple virtual machines, each of which may host multiple LGTM workers. In a Dockerized deployment, each worker daemon runs in a separate Docker container.

> **Important**
> It's important to distinguish between "workers" (the daemons that build and analyze code) and "worker hosts" (the machines on which the workers run).
>
> The workers in LGTM's work pool should not be confused with the "task workers" that run in the control pool. The latter are instances of a service that performs background tasks for the system.

Each worker can access three types of resource:

- **Repositories**
  Workers can download source code from a repository.
- **File store**
  Workers download files from the file store in the control pool. Files include Semmle Core files and project configuration files. After analysis, workers upload data to the file store—for example, CodeQL databases for a project, which also contain query results and a source archive.
- **Job dealer**
  Workers poll the job dealer for the next job to work on. Some workers are configured only to retrieve jobs that run user-defined queries from the Query console, to ensure these are processed quickly. If a worker has one or more labels assigned to it, it retrieves jobs with matching labels in preference to unlabeled jobs, and ignores any jobs with other labels. On completion of a job, the worker informs the job dealer.

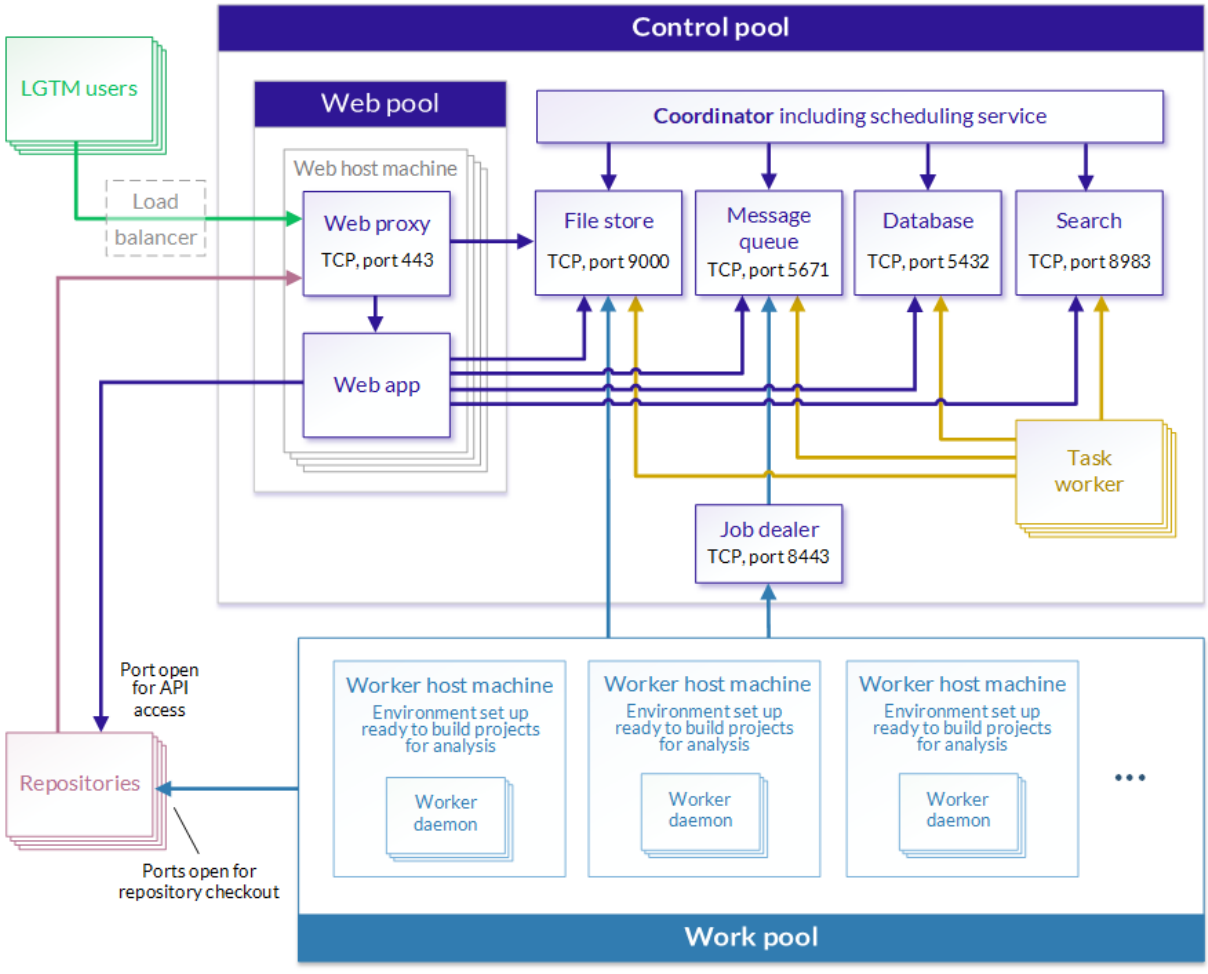


Note
The arrow directions on the diagram give an indication of data flow between system components.

Network connections

For a live site deployment, the system is scaled by being installed on a server cluster containing control pool and work pool machines and, optionally, a separate web pool. When the web pool contains more than one machine, you can use an external load balancer, or round-robin DNS, to share the load between the web pool machines.

> **Note**
> In the following diagram, the arrow directions indicate connection requests from one system component to the component to which the arrow points. Where an arrow points from A to B, A must be able to see B, and B must accept requests from A on the specified port.



Port availability

The diagram above shows the ports on which parts of the system listen for connections. These parts of the system can be hosted on different machines, as required. Depending on your cluster topology, you must make sure that the appropriate ports are reachable on the machines in the cluster, based on the LGTM services that are running on each machine. The following basic cluster topologies detail which TCP ports must be reachable. For other topologies, see the diagram above.

Single machine

Running the whole system on a single machine is not appropriate for a live system, but for a demo system the following port must be reachable:

Port	Must be reachable by
443	Users' browsers and repository host systems

Separate control pool and work pool

Where all elements of the control pool are hosted on a single "controller" machine, with the work pool hosted on one or more separate machines, the following ports must be reachable on the **controller machine**:

Port	Must be reachable by
443	Users' browsers and repository host systems
8443	The worker host machine(s)—for connections to the job dealer service
9000	The worker host machine(s)—for connections to the file store service

Separate control, web, and work pools

Where the web pool is hosted separately from the "controller" machine (which hosts all other elements of the control pool), and the work pool is also hosted on one or more separate machines, the following port must be reachable on the **machine that hosts the web proxy**:

Port	Must be reachable by
443	Users' browsers and repository host systems Optionally the load balancer (if you are running multiple instances of the web proxy and web app)

And these ports must be reachable on the **controller machine**:

Port	Must be reachable by
8443	The worker host machine(s)—for connections to the job dealer service
9000	The worker host machine(s) and the web pool machine(s)—for connections to the file store service
5671	The web pool machine(s)—for connections to the message queue service
5432	The web pool machine(s)—for connections to the database service
8983	The web pool machine(s)—for connections to the search service

Communication security

All LGTM components use secure channels, protected by SSL certificates, to communicate with each other. LGTM also uses secure connections to fetch source code from your repositories, provided that you configure the repository host and LGTM to use secure connections. For more information, see Securing LGTM Enterprise.

Worker processes

The architecture of the workers in LGTM's work pool is similar to that of continuous integration tools like Jenkins and Atlassian Bamboo. That is:

- No barriers are enforced between builds running on the same machine, either concurrently or consecutively.
- A single operating system user name is used for multiple build and analysis tasks.
- Data is cached between tasks on LGTM workers to improve performance. This is principally the data extraction tools, but may also include the credentials required to access one or more repositories.

- The LGTM configuration for a repository, which controls the build and analysis process, is stored in the repository.

If you're concerned about securing one or more codebases from potential attacks by your own developers, deploy a separate instance of LGTM for these projects. Set this instance up to run its worker daemons on machines that serve exclusively as worker hosts for this LGTM.