Managing self-hosted runners for Dependabot updates on your enterprise

In this article

You can create dedicated runners for your GitHub Enterprise Server instance that Dependabot uses to create pull requests to help secure and maintain the dependencies used in repositories on your enterprise.

About self-hosted runners for Dependabot updates

You can help users of your GitHub Enterprise Server instance to create and maintain secure code by setting up Dependabot security and version updates. With Dependabot updates, developers can configure repositories so that their dependencies are updated and kept secure automatically. For more information, see "Enabling Dependabot for your enterprise."

To use Dependabot updates on your GitHub Enterprise Server instance, you must configure self-hosted runners to create the pull requests that will update dependencies.

Prerequisites

Configuring self-hosted runners is only one step in the middle of the process for enabling Dependabot updates. There are several steps you must follow before these steps, including configuring your GitHub Enterprise Server instance to use GitHub Actions with self-hosted runners. For more information, see "Enabling Dependabot for your enterprise."

Configuring self-hosted runners for Dependabot updates

After you configure your GitHub Enterprise Server instance to use GitHub Actions, you need to add self-hosted runners for Dependabot updates.

System requirements for Dependabot runners

Any VM that you use for Dependabot runners must meet the requirements for self-hosted runners. In addition, they must meet the following requirements.

- Linux operating system

- x64 architecture

- Docker installed with access for the runner users:
  - We recommend installing Docker in rootless mode and configuring the runners to access Docker without `root` privileges.
  - Alternatively, install Docker and give the runner users raised privileges to run Docker.

The CPU and memory requirements will depend on the number of concurrent runners you deploy on a given VM. As guidance, we have successfully set up 20 runners on a single 2 CPU 8GB machine, but ultimately, your CPU and memory requirements will heavily depend on the repositories being updated. Some ecosystems will require more resources than others.

If you specify more than 14 concurrent runners on a VM, you must also update the Docker `/etc/docker/daemon.json` configuration to increase the default number of networks Docker can create.

```
{
  "default-address-pools": [
    {"base":"10.10.0.0/16","size":24}
  ]
}
```

Network requirements for Dependabot runners 🔗

Dependabot runners require access to the public internet, GitHub.com, and any internal registries that will be used in Dependabot updates. To minimize the risk to your internal network, you should limit access from the Virtual Machine (VM) to your internal network. This reduces the potential for damage to internal systems if a runner were to download a hijacked dependency.

Adding self-hosted runners for Dependabot updates 🔗

1 Provision self-hosted runners, at the repository, organization, or enterprise account level. For more information, see "About self-hosted runners" and "Adding self-hosted runners."

2 Set up the self-hosted runners with the requirements described above. For example, on a VM running Ubuntu 20.04 you would:

- Install Docker and ensure that the runner users have access to Docker. For more information, see the Docker documentation.

  - Install Docker Engine on Ubuntu
  - Recommended approach: Run the Docker daemon as a non-root user (Rootless mode)
  - Alternative approach: Manage Docker as a non-root user

- Verify that the runners have access to the public internet and can only access the internal networks that Dependabot needs.

3 Assign a `dependabot` label to each runner you want Dependabot to use. For more information, see "Using labels with self-hosted runners."

4 Optionally, enable workflows triggered by Dependabot to use more than read-only permissions and to have access to any secrets that are normally available. For more information, see "Troubleshooting GitHub Actions for your enterprise."

Legal

Terms   Privacy   Status   Pricing   Expert services   Blog