

This version of GitHub Enterprise was discontinued on 2023-03-15. No patch releases will be made, even for critical security issues. For better performance, improved security, and new features, [upgrade to the latest version of GitHub Enterprise](#). For help with the upgrade, [contact GitHub Enterprise support](#).

Installing GitHub Enterprise Server on Azure

In this article

Prerequisites

Hardware considerations

Determining the virtual machine type

Creating the GitHub Enterprise Server virtual machine

Configuring the GitHub Enterprise Server virtual machine

Azure extension features

Further reading

To install GitHub Enterprise Server on Azure, you must deploy onto a memory-optimized instance that supports premium storage.

You can deploy GitHub Enterprise Server on global Azure or Azure Government.

Prerequisites

- You must have a GitHub Enterprise license file. For more information, see "[Setting up a trial of GitHub Enterprise Server](#)" and "[About licenses for GitHub Enterprise](#)."
- You must have an Azure account capable of provisioning new machines. For more information, see the [Microsoft Azure website](#).
- Most actions needed to launch your virtual machine (VM) may also be performed using the Azure Portal. However, we recommend installing the Azure command line interface (CLI) for initial setup. Examples using the Azure CLI 2.0 are included below. For more information, see Azure's guide "[Install Azure CLI 2.0](#)."

Hardware considerations

- [Minimum requirements](#)
- [Storage](#)
- [CPU and memory](#)

Minimum requirements

We recommend different hardware configurations depending on the number of user licenses for your GitHub Enterprise Server instance. If you provision more resources than the minimum requirements, your instance will perform and scale better.

User licenses	x86-64 vCPUs	Memory	Root storage	Attached (data) storage
Trial, demo, or 10 light users	4	32 GB	200 GB	150 GB
10 to 3,000	8	48 GB	200 GB	300 GB
3,000 to 5000	12	64 GB	200 GB	500 GB
5,000 to 8000	16	96 GB	200 GB	750 GB
8,000 to 10,000+	20	160 GB	200 GB	1000 GB

If you plan to enable GitHub Actions for the users of your instance, more resources are required.

vCPUs	Memory	Maximum Concurrency
8	64 GB	300 jobs
16	128 GB	700 jobs
32	160 GB	1500 jobs
64	256 GB	2500 jobs
96	384 GB	4500 jobs
128	550 GB	5000 jobs

For more information about these requirements, see "[Getting started with GitHub Actions for GitHub Enterprise Server](#)."

If you plan to enable Container registry for the users of your instance, more resources are required. For more information about these requirements, see "[Getting started with GitHub Packages for your enterprise](#)."

For more information about adjusting resources for an existing instance, see "[Increasing storage capacity](#)" and "[Increasing CPU or memory resources](#)."

Storage

We recommend a high-performance SSD with high input/output operations per second (IOPS) and low latency for GitHub Enterprise Server. Workloads are I/O intensive. If you use a bare metal hypervisor, we recommend directly attaching the disk or using a disk from a storage area network (SAN).

Your instance requires a persistent data disk separate from the root disk. For more information, see "[System overview](#)."

To configure GitHub Actions, you must provide external blob storage. For more information, see "[Getting started with GitHub Actions for GitHub Enterprise Server](#)."

The available space on the root filesystem will be 50% of the total disk size. You can resize your instance's root disk by building a new instance or using an existing instance. For more information, see "[System overview](#)" and "[Increasing storage capacity](#)."

CPU and memory

The CPU and memory resources that GitHub Enterprise Server requires depend on the

levels of activity for users, automations, and integrations.

Any VMs you provision for your GitHub Enterprise Server instance must use the x86-64 CPU architecture. Other architectures are not supported, such as Aarch64 or arm64.

If you plan to enable GitHub Actions for the users of your GitHub Enterprise Server instance, you may need to provision additional CPU and memory resources for your instance. For more information, see "[Getting started with GitHub Actions for GitHub Enterprise Server](#)."

When you increase CPU resources, we recommend adding at least 6.5 GB of memory for each vCPU (up to 16 vCPUs) that you provision for the instance. When you use more than 16 vCPUs, you don't need to add 6.5 GB of memory for each vCPU, but you should monitor your instance to ensure it has enough memory.

Warning: We recommend that users configure webhook events to notify external systems of activity on GitHub Enterprise Server. Automated checks for changes, or *polling*, will negatively impact the performance and scalability of your instance. For more information, see "[About webhooks](#)."

For more information about monitoring the capacity and performance of GitHub Enterprise Server, see "[Monitoring your appliance](#)."

You can increase your instance's CPU or memory resources. For more information, see "[Increasing CPU or memory resources](#)."

Determining the virtual machine type

Before launching your GitHub Enterprise Server instance on Azure, you'll need to determine the machine type that best fits the needs of your organization. For more information about memory optimized machines, see "[Memory optimized virtual machine sizes](#)" in the Microsoft Azure documentation. To review the minimum resource requirements for GitHub Enterprise Server, see "[Minimum requirements](#)."

Note: You can always scale up your CPU or memory by resizing your instance. However, because resizing your CPU or memory requires downtime for your users, we recommend over-provisioning resources to account for scale.

The GitHub Enterprise Server appliance requires a premium storage data disk, and is supported on any Azure VM that supports premium storage. Azure VM types with the **s** suffix support premium storage. For more information, see "[What disk types are available in Azure?](#)" and "[Azure premium storage: design for high performance](#)" in the Azure documentation.

GitHub recommends a memory-optimized VM for GitHub Enterprise Server. For more information, see "[Memory optimized virtual machine sizes](#)" in the Azure documentation.

GitHub Enterprise Server supports any region that supports your VM type. For more information about the supported regions for each VM, see Azure's "[Products available by region](#)."

Creating the GitHub Enterprise Server virtual machine

To create the instance, you'll need to import the GitHub Enterprise Server image to your virtual machine and attach an additional storage volume for your instance data. For more information, see "[Hardware considerations](#)."

- 1 Find the most recent GitHub Enterprise Server appliance image. For more information about the `vm image list` command, see "[az vm image list](#)" in the Microsoft documentation.

```
$ az vm image list --all -f GitHub-Enterprise | grep '"urn":' | sort -V
```

- 2 Create a new VM using the appliance image you found. For more information, see "[az vm create](#)" in the Microsoft documentation.

Pass in options for the name of your VM, the resource group, the size of your VM, the name of your preferred Azure region, the name of the appliance image VM you listed in the previous step, and the storage SKU for premium storage. For more information about resource groups, see "[Resource groups](#)" in the Microsoft documentation.

```
$ az vm create -n VM_NAME -g RESOURCE_GROUP --size VM_SIZE -l REGION --image AP
```

- 3 Configure the security settings on your VM to open up required ports. For more information, see "[az vm open-port](#)" in the Microsoft documentation. See the table below for a description of each port to determine what ports you need to open.

```
$ az vm open-port -n VM_NAME -g RESOURCE_GROUP --port PORT_NUMBER
```

This table identifies what each port is used for.

Port	Service	Description
22	SSH	Git over SSH access. Clone, fetch, and push operations to public/private repositories supported.
25	SMTP	SMTP with encryption (STARTTLS) support.
80	HTTP	Web application access. <i>All requests are redirected to the HTTPS port when SSL is enabled.</i>
122	SSH	Instance shell access. <i>The default SSH port (22) is dedicated to application git+ssh network traffic.</i>
161/UDP	SNMP	Required for network monitoring protocol operation.
443	HTTPS	Web application and Git over HTTPS access.
1194/UDP	VPN	Secure replication network tunnel in high availability configuration.
8080	HTTP	Plain-text web based Management Console. <i>Not required unless SSL is</i>

disabled manually.

8443	HTTPS	Secure web based Management Console. <i>Required for basic installation and configuration.</i>
9418	Git	Simple Git protocol port. Clone and fetch operations to public repositories only. <i>Unencrypted network communication.</i> If you have enabled private mode on your instance, then opening this port is only required if you also enabled anonymous Git read access. For more information, see " Enforcing repository management policies in your enterprise ."

- 4 Create and attach a new unencrypted data disk to the VM, and configure the size based on your user license count. For more information, see "[az vm disk attach](#)" in the Microsoft documentation.

Pass in options for the name of your VM (for example, `ghe-acme-corp`), the resource group, the premium storage SKU, the size of the disk (for example, `200`), and a name for the resulting VHD.

```
$ az vm disk attach --vm-name VM_NAME -g RESOURCE_GROUP --sku Premium_LRS --new
```

Note: For non-production instances to have sufficient I/O throughput, the recommended minimum disk size is 150 GiB with read/write cache enabled (`--caching ReadWrite`).

Configuring the GitHub Enterprise Server virtual machine [↗](#)

To configure the instance, you must confirm the instance's status, upload a license file, set the Management Console password, configure the instance's settings, and restart the instance.

Warning: To prevent an attacker from compromising the new instance, ensure that you personally set the Management Console password and create the first user as soon as possible.

- 1 Before configuring the VM, you must wait for it to enter ReadyRole status. Check the status of the VM with the `vm list` command. For more information, see "[az vm list](#)" in the Microsoft documentation.

```
$ az vm list -d -g RESOURCE_GROUP -o table
> Name      ResourceGroup  PowerState  PublicIps      Fqdns      Location  Zc
> -----
> VM_NAME RESOURCE_GROUP  VM running  40.76.79.202   eastus
```

Note: Azure does not automatically create a FQDNS entry for the VM. For more information, see Azure's guide on how to "[Create a fully qualified domain name in the Azure portal for a Linux VM](#)."

- a. Copy the virtual machine's public DNS name, and paste it into a web browser.
- b. At the prompt, upload your license file and set a management console password. For more information, see "[Managing your license for GitHub Enterprise](#)."
- c. In the [Management Console](#), configure and save your desired settings. For more information, see "[Configuring your enterprise](#)."
- d. The instance will restart automatically.
- e. Click **Visit your instance**.

Azure extension features

GitHub Enterprise Server does not support the installation of Azure extension features. The GitHub Enterprise Server image is shipped with a customized `waagent` package which only supports basic VM management functions and blocks advanced VM management functions.

To avoid system instability of your GitHub Enterprise Server instance, the `walinuxagent` service is intentionally run in GitHub Enterprise Server in a restricted mode, explicitly disallowing the agent from being able to install other agents. VM management features that rely on additional agents and extensions beyond that which ships with GitHub Enterprise Server image, such as the Monitoring Agent extension for Azure Insights or Azure Backups, are unsupported.

Because GitHub Enterprise Server runs a customized Linux operating system with only the necessary applications and services, installing or updating operating system packages manually will overwrite these customizations and can cause unexpected behavior. For more information, see "[System overview](#)."

Further reading

- "[System overview](#)"
- "[About upgrades to new releases](#)"

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)