Configuring SSH connections to your instance

In this article

You can increase the security of your GitHub Enterprise Server instance by configuring the SSH algorithms that clients can use to establish a connection.

> **Who can use this feature**
> Site administrators can configure SSH connections to a GitHub Enterprise Server instance.

About SSH connections to your instance

Each GitHub Enterprise Server instance accepts SSH connections over two ports. Site administrators can access the administrative shell via SSH, then run command-line utilities, troubleshoot, and perform maintenance. Users can connect via SSH to access and write Git data in the instance's repositories. Users do not have shell access to your instance. For more information, see the following articles.

- "Network ports"
- "Accessing the administrative shell (SSH)"
- "About SSH"

To accommodate the SSH clients in your environment, you can configure the types of connections that your GitHub Enterprise Server instance will accept.

Configuring SSH connections with RSA keys

When users perform Git operations on your GitHub Enterprise Server instance via SSH over port 22, the client can authenticate with an RSA key. The client may sign the attempt using the SHA-1 hash function. In this context, the SHA-1 hash function is no longer secure. For more information, see SHA-1 on Wikipedia.

By default, SSH connections that satisfy **both** of the following conditions will fail.

- The RSA key was added to a user account on your GitHub Enterprise Server instance after the cutoff date of midnight UTC on August 1, 2022.
- The SSH client signs the connection attempt with the SHA-1 hash function.

You can adjust the cutoff date. If the user uploaded the RSA key before the cutoff date, the client can continue to connect successfully using SHA-1 as long as the key remains valid. Alternatively, you can reject all SSH connections authenticated with an RSA key if the client signs the connection using the SHA-1 hash function.

Regardless of the setting you choose for your instance, clients can continue to connect

using any RSA key signed with a SHA-2 hash function.

If you use an SSH certificate authority, connections will fail if the certificate's `valid_after` date is after the cutoff date. For more information, see "About SSH certificate authorities."

For more information, see the GitHub Blog.

1. SSH into your GitHub Enterprise Server instance. If your instance comprises multiple nodes, for example if high availability or geo-replication are configured, SSH into the primary node. If you use a cluster, you can SSH into any node. For more information about SSH access, see "Accessing the administrative shell (SSH)."

   ```
   ssh -p 122 admin@HOSTNAME
   ```

2. Audit your instance's logs for connections that use unsecure algorithms or hash functions using the `ghe-find-insecure-git-operations` utility. For more information, see "Command-line utilities."

3. To configure a cutoff date after which your GitHub Enterprise Server instance will deny connections from clients that use an RSA key uploaded after the date if the connection is signed by the SHA-1 hash function, enter the following command. Replace ***RFC-3399-UTC-TIMESTAMP*** with a valid RFC 3399 UTC timestamp. For example, the default value, August 1, 2022, would be represented as `2022-08-01T00:00:00Z`. For more information, see RFC 3339 on the IETF website.

   ```
   $ ghe-config app.gitauth.rsa-sha1 RFC-3339-UTC-TIMESTAMP
   ```

4. Alternatively, to completely disable SSH connections using RSA keys that are signed with the SHA-1 hash function, enter the following command.

   ```
   ghe-config app.gitauth.rsa-sha1 false
   ```

5. To apply the configuration, run the following command.

   Note: During a configuration run, services on your GitHub Enterprise Server instance may restart, which can cause brief downtime for users.

   Shell

   ```
   ghe-config-apply
   ```

6. Wait for the configuration run to complete.