

Auditing SSH keys

In this article

- Initiating an audit
- What users see
- Adding an SSH key

Site administrators can initiate an instance-wide audit of SSH keys.

Once initiated, the audit disables all existing SSH keys and forces users to approve or reject them before they're able to clone, pull, or push to any repositories. An audit is useful in situations where an employee or contractor leaves the company and you need to ensure that all keys are verified.

Initiating an audit

You can initiate an SSH key audit from the "All users" tab of the site admin dashboard. After you click the **Start public key audit** button, you'll be taken to a confirmation screen explaining that initiating an SSH key audit will disable all public keys, preventing pushing and pulling over SSH. Users will be required to verify their public keys to restore SSH access.

After you click the **Begin audit** button, all SSH keys are invalidated and will require approval. You'll see a notification indicating the audit has begun.

What users see

If a user attempts to perform any git operation over SSH, it will fail and provide them with the following message:

```
ERROR: Hi USERNAME. We're doing an SSH key audit.
Please visit http(s)://HOSTNAME/settings/ssh/audit/2
to approve this key so we know it's safe.
Fingerprint: ed:21:60:64:c0:dc:2b:16:0f:54:5f:2b:35:2a:94:91
fatal: The remote end hung up unexpectedly
```

When they follow the link, they're asked to approve the keys on their account. After they approve or reject their keys, they'll be able to interact with repositories as usual.

Adding an SSH key

When a new user adds an SSH key to an account, to confirm the user's access, GitHub Enterprise Server will prompt for authentication. For more information, see "[Sudo mode](#)."

When a user adds a key, they'll receive a notification email that will look something like this:

The following SSH key was added to your account:

[title]

ed:21:60:64:c0:dc:2b:16:0f:54:5f:2b:35:2a:94:91

If you believe this key was added in error, you can remove the key and disable access at the following location:

http(s)://HOSTNAME/settings/ssh

Legal

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