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.

Adding a GPG key to your GitHub account

In this article

To configure your account on your GitHub Enterprise Server instance to use your new (or existing) GPG key, you'll also need the key to your account.

About addition of GPG keys to your account

To sign commits associated with your account on GitHub Enterprise Server, you can add a public GPG key to your personal account. Before you add a key, you should check for existing keys. If you don't find any existing keys, you can generate and copy a new key. For more information, see "Checking for existing GPG keys" and "Generating a new GPG key."

You can add multiple public keys to your account on GitHub Enterprise Server. Commits signed by any of the corresponding private keys will show as verified. If you remove a public key, any commits signed by the corresponding private key will no longer show as verified.

Supported GPG key algorithms

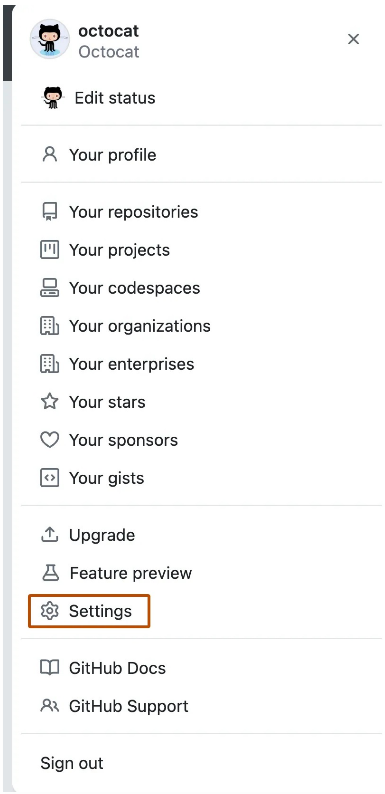
GitHub Enterprise Server supports several GPG key algorithms. If you try to add a key generated with an unsupported algorithm, you may encounter an error.

- RSA
- ElGamal
- DSA
- ECDH
- ECDSA
- EdDSA

When verifying a signature, GitHub Enterprise Server extracts the signature and attempts to parse its key ID. The key ID is then matched with keys added to GitHub Enterprise Server. Until a matching GPG key is added to GitHub Enterprise Server, it cannot verify your signatures.

Adding a GPG key

1. In the upper-right corner of any page, click your profile photo, then click **Settings**.



2. In the user settings sidebar, click **SSH and GPG keys**.
3. Next to the "GPG keys" header, click **New GPG key**.
4. In the "Title" field, type a name for your GPG key.
5. In the "Key" field, paste the GPG key you copied when you generated your GPG key.
6. Click **Add GPG key**.
7. To confirm the action, authenticate to your GitHub account.

Updating an expired GPG key

When verifying a signature, GitHub Enterprise Server checks that the key is not revoked or expired. If your signing key is revoked or expired, GitHub Enterprise Server cannot verify your signatures.

If your key is expired, you must update its expiration, export the new key, delete the expired key in your account on GitHub Enterprise Server, and add the new key to your account as described above. Your previous commits and tags will show as verified, as long as the key meets all other verification requirements.

If your key is revoked, use the primary key or another key that is not revoked to sign your commits.

If your key is invalid and you don't use another valid key in your key set, but instead generate a new GPG key with a new set of credentials, then your commits made with the revoked or expired key will continue to show as unverified. Also, your new credentials will not be able to re-sign or verify your old commits and tags.

Further reading

- "Checking for existing GPG keys"
- "Generating a new GPG key"
- "Telling Git about your signing key"
- "Associating an email with your GPG key"
- "Signing commits"
- "About commit signature verification"