HTTPS setup to encrypt connections to Gitea

**Table of Contents**



# Using the built-in server

Before you enable HTTPS, make sure that you have valid SSL/TLS certificates. You could use self-generated certificates for evaluation and testing. Please run `gitea cert --host [HOST]` to generate a self signed certificate.
If you are using Apache or nginx on the server, it’s recommended to check the reverse proxy guide.
To use Gitea’s built-in HTTPS support, you must change your app.ini file:

```
[server]
PROTOCOL  = https
ROOT_URL  = https://git.example.com:3000/
HTTP_PORT = 3000
CERT_FILE = cert.pem
KEY_FILE  = key.pem
```

Note that if your certificate is signed by a third party certificate authority (i.e. not self-signed), then cert.pem should contain the certificate chain. The server certificate must be the first entry in cert.pem, followed by the intermediaries in order (if any). The root certificate does not have to be included because the connecting client must already have it in order to estalbish the trust relationship. To learn more about the config values, please checkout the Config Cheat Sheet.

## Setting up HTTP redirection

The Gitea server is only able to listen to one port; to redirect HTTP requests to the HTTPS port, you will need to enable the HTTP redirection service:

```
[server]
REDIRECT_OTHER_PORT = true
; Port the redirection service should listen on
PORT_TO_REDIRECT = 3080
```

If you are using Docker, make sure that this port is configured in your docker-compose.yml file.

# Using Let’s Encrypt

Let’s Encrypt is a Certificate Authority that allows you to automatically request and renew SSL/TLS certificates. In addition to starting Gitea on your configured port, to request HTTPS certificates, Gitea will also need to listed on port 80, and will set up an autoredirect to HTTPS for you. Let’s Encrypt will need to be able to access Gitea via the Internet to verify your ownership of the domain.

By using Let’s Encrypt **you must consent** to their terms of service.

```
[server]
PROTOCOL=https
DOMAIN=git.example.com
ENABLE_LETSENCRYPT=true
LETSENCRYPT_ACCEPTTOS=true
LETSENCRYPT_DIRECTORY=https
LETSENCRYPT_EMAIL=email@example.com
```

To learn more about the config values, please checkout the Config Cheat Sheet.

## Using a reverse proxy

Setup up your reverse proxy as shown in the reverse proxy guide.
After that, enable HTTPS by following one of these guides:

- nginx
- apache2/httpd
- caddy

Note: Enabling HTTPS only at the proxy level is referred as TLS Termination Proxy. The proxy server accepts incoming TLS connections, decrypts the contents, and passes the now unencrypted contents to Gitea. This is normally fine as long as both the proxy and Gitea instances are either on the same machine, or on different machines within private network (with the proxy is exposed to outside network). If your Gitea instance is separated from your proxy over a public network, or if you want full end-to-end encryption, you can also enable HTTPS support directly in Gitea using built-in server and forward the connections over HTTPS instead.