



EN



donate to EFF



# certbot instructions

what's your http website running on?

Apache

Ubuntu 20

[Help, I'm not sure!](#)

## Apache on Ubuntu 20

default

wildcard

to use certbot, you'll need...



comfort with the [command line](#)

**nnp://**

that is already online  
with an open port 80

EN



...which is hosted on a server  
which you can access via ssh  
with the ability to sudo  
*optional if you want a wildcard cert : DNS credentials*

## Snap Support

The Certbot snap supports the x86\_64, ARMv7, and ARMv8 architectures. While we strongly recommend that most users install Certbot through the snap, you can find alternate installation instructions [here](#).

### 1. SSH into the server

SSH into the server running your HTTP website as a user with sudo privileges.

### 2. Install snapd

You'll need to install snapd and make sure you follow any instructions to enable classic snap support.

Follow these instructions on [snapcraft's site to install snapd](#).

**install snapd**

If you have any Certbot packages installed using an OS package manager like `apt`, `dnf`, or `yum`, you should remove them before installing the Certbot snap to ensure that when you run the command `certbot` the snap is used rather than the installation from your OS package manager. The exact command to do this depends on your OS, but common examples are `sudo apt-get remove certbot`, `sudo dnf remove certbot`, or `sudo yum remove certbot`.

## 4. Install Certbot

Run this command on the command line on the machine to install Certbot.

```
$ sudo snap install --classic certbot
```

## 5. Prepare the Certbot command

Execute the following instruction on the command line on the machine to ensure that the `certbot` command can be run.

```
$ sudo ln -s /snap/bin/certbot /usr/bin/certbot
```

## 6. Choose how you'd like to run Certbot

### ***Either get and install your certificates...***

Run this command to get a certificate and have Certbot edit your apache configuration automatically to serve it, turning on HTTPS access in a single step.

```
$ sudo certbot --apache
```

### ***Or just get a certificate***

If you're feeling more conservative and would like to make the changes to your apache configuration by hand, run this command:

**EN**

```
$ sudo certbot certonly --apache
```

## 7. Test automatic renewal

The Certbot packages on your system come with a cron job or systemd timer that will renew your certificates automatically before they expire. You will not need to run Certbot again, unless you change your configuration. You can test automatic renewal for your certificates by running this command:

```
$ sudo certbot renew --dry-run
```

The command to renew certbot is installed in one of the following locations:

- `/etc/crontab/`
- `/etc/cron.*/*`
- `systemctl list-timers`

## 8. Confirm that Certbot worked

To confirm that your site is set up properly, visit `https://yourwebsite.com/` in your browser and look for the lock icon in the URL bar.