

# How To: Enable Password-less SUDO.

This isn't the smartest thing you can do. In fact, you probably shouldn't do this. But, if you are comfortable with your physical security, you can use sudo without a password.

In my case, there's not a whole lot folks are going to do with sudo on my computer. Anyone with physical access to my device is someone that I trust. I also run a ton of commands when hanging out in the support sites and I am frankly just tired of typing my password when I use sudo.

So, let's get rid of it. Start by pressing CTRL + ALT + T, and then enter:

```
[code]sudo nano /etc/sudoers[/code]
```

Scroll down to the bottom and add this line:

```
[code]<your_username> ALL=(ALL) NOPASSWD:ALL[/code]
```

Where "<your\_username>" substitute it with your actual username on your computer. Now save it with:

CTRL + X

Y

ENTER

See that? You also may have just learned how use 'nano' to edit and save a text file while in the terminal. Pretty neat, huh? Anyhow, scroll up a little and look to the right. There's a spot where you can enter a name and email address. If you do that (and confirm the email address) then you'll get handy notices in the email when there's a new article. I promise, I won't send you a single non-site related email – ever.

---

# Use `.htaccess` to Stop People From Viewing Files in a Directory.

Open the directory with your favorite FTP client – unless you're local and can just navigate to the directory.

Add a file called `.htaccess`. The `'.'` is important and mandatory.

The permissions for `.htaccess` should be 644.

Edit the `.htaccess` file with a plain-text editor (don't use a word-processor application) and add the following line:

```
[code]Options -Indexes[/code]
```

Save the file.

What this will do is prevent indexing the files in the folder. If people try to access the folder directly, they'll get a 403 forbidden error.

At the same time, you can still link directly to files in that folder.

So, let's say you added the `.htaccess` to a directory called `/tmp`. You can still link to, use, and send people to `/tmp/picture.jpg` like normal, but people won't be able to browse the directory and find files you don't want them to see.

For more information, [click this](#).

---

# HSTS Preload

Seeing as the site is hosted on a Linux server, I'll go ahead and document this. This is how to set up HSTS Preload for your website, by using `.htaccess`.

If you want to get your site hard-coded into Google Chrome as an "HTTPS Only" site, it's actually relatively easy – once you know how to do it. You can verify that this site is listed [here](#).

Open your site with your favorite FTP application, I prefer Filezilla. Make sure that you've set it to show hidden files. (Files prefaced with a period are hidden files by default.) Edit your `.htaccess` and add the following:

```
[code]<IfModule mod_headers.c>
Header set Strict-Transport-Security "max-age=31536000;
includeSubDomains; preload" env=HTTPS
</IfModule>[/code]
```

Save your file to your server and check your site for yourself.

Why do this? It's added security for your visitors and it's quite probable that Google has a preference for sites who have taken the time to do so. That may lead to more traffic and happier traffic because they know your site is using HTTPS.