

# RewriteHTTPToHTTPS

## HTTP to HTTPS

### Scenario :

You want to force people coming to your site to use HTTPS. Either for the entire site or a small sub-section of it.

- Note\*

Using `mod_rewrite` to do this isn't the recommended behavior. See [RedirectSSL](#)

### Fix :

```
RewriteEngine On
# This will enable the Rewrite capabilities

RewriteCond %{HTTPS} !=on
# This checks to make sure the connection is not already HTTPS

RewriteRule ^/?(.*?) https://%{SERVER_NAME}/$1 [R,L]
# This rule will redirect users from their original location, to the same location but using HTTPS.
# i.e. http://www.example.com/foo/ to https://www.example.com/foo/
# The leading slash is made optional so that this will work either in httpd.conf
# or .htaccess context
```

### Entire site (.htaccess) :

*Note: While the rules you need are the same as above (because the rule above doesn't depend on any of the quirks of rewrite in .htaccess), you will need to ensure that you place this in a .htaccess file in the root of the site you want to apply it against, and to make sure you have the appropriate AllowOverride configuration in your httpd.conf*

### Specific Directory

*Either put the above solution in a .htaccess file in the directory to be affected, or put the URI prefix in the regex itself.*

```
RewriteEngine On
# This will enable the Rewrite capabilities

RewriteCond %{HTTPS} !=on
# This checks to make sure the connection is not already HTTPS

RewriteRule ^/?secure/(.*) https://%{SERVER_NAME}/secure/$1 [R,L]
# This rule will redirect all users who are using any part of /secure/ to the same location but using HTTPS.
# i.e. http://www.example.com/secure/ to https://www.example.com/secure/
# This means if you dont want to force HTTPS for all directories you can force it for a specific sub-section of
the site.
```