

# RewriteImageTheft

## Problem

The general idea here is that you want to prevent people from including your images in their webpages. This is termed "image theft" or "bandwidth theft" simply because they are using your resources for their website. The following rulesets will prevent that from happening, based on the HTTP\_REFERER header. There are two main ways that you might wish to deal with this. You can either simply forbid the request, as the first recipe does, or you can redirect that request to some other resource, such as a "please go away" image, or perhaps something even less appealing, as you like.

## Forbid the request

In this example, **example.com** should, of course, be replaced with your own website address.

```
RewriteEngine on
RewriteCond %{HTTP_REFERER} !=""
RewriteCond %{HTTP_REFERER} !example\.com [NC]
RewriteRule \.(jpe?g|gif|png)$ - [F,NC]
```

The result of this ruleset is that any requests that aren't referred by pages on **example.com** or **something.example.com** will generate a 403 Forbidden message, resulting in a broken image on the other end.

Note that the referrer may also be blank, which allows for clients which do not pass a referrer for some reason.

## Replace the request

```
RewriteEngine on
RewriteCond %{HTTP_REFERER} !=""
RewriteCond %{HTTP_REFERER} !example\.com [NC]
# depending upon in which context you use the RewriteRule,
# you might need a condition to exclude the go_away.png to prevent
# an internal redirect looping. We don't use a RegEx here::
RewriteCond %{REQUEST_URI} !=/images/go_away.png
RewriteRule \.(jpe?g|gif|png)$ /images/go_away.png [NC,L]
```

If you want the image to be replaced with a URL from another site, you can do so by providing a full URL in that RewriteRule:

```
RewriteRule \.(jpe?g|gif|png)$ http://other.example.com/images/go_away.png [R,NC,L]
```

## Discussion

The idea here is to only allow your own domains, and those you authorize, to be able to hot-link to your site's images.

This may be extended to multiple approved hostnames by adding additional RewriteCond directives (or modifying the regular expression) enumerating those hostnames.

We also permit the HTTP\_REFERER to be empty (="") so that if something is requested directly (by typing in a URL) or by a client that doesn't pass REFERER information (e.g. caused by some proxy cacheing machines, firewalls, privacy software) the request will still be permitted.

If the RewriteConds match (ie, the request is not one of the required values) the the request will instead be redirected to another page, explaining why image theft is a bad idea. You could alternately redirect to another image, or simply refuse the connection using the [F] or [R=403] (Apache 2.1 and later) flag.