

Understanding Redirects

- [Introduction](#)
- [Browser / Client Side Redirects](#)
- [Server Side Redirects](#)
- [Redirects in Microsoft Internet Explorer](#)

Introduction

Redirects provide a way to transport the browser from one point on a web site to another. Redirects are most commonly used to translate references to outdated web pages to new, updated ones. They can be instigated either by the browser, in which case they are referred to as client-side redirect, or by the server, in which case they are referred to as server-side redirects.

Virtually all webmasters at some point will discover that not all links to their site end up where they were intended. For example, if the author of another web site places a link to your site, but misspells the hyperlink, the result will be a 404 error every time that link is selected. Another common occurrence is site reorganisations. As you move files around, rename them or even delete them entirely, you will find visitors will receive 404 errors. This is because other sites, search engines and directories often link to those pages. It is an impossible task to be able to change all the hyperlinks to your site. If the webmaster determines that someone has linked to a non-existent page, there are ways of redirecting the user to the correct page.

Browser / Client Side Redirects

If you are a webmaster and you wish to recapture the traffic from the lost/moved/deleted pages, there are several things that can be done. One of the easiest ways to add a redirect is through a special Meta tag called "refresh" to direct their visitors to the new page. To do this they create a blank page which contains the tag as shown in Figure 1 (it is placed in the header). The tag simply says "wait some time then go to a named page". In this example, the browser will wait one second.

HTML Client-Side Redirect

```
<META HTTP-EQUIV="refresh" content= "1;URL=http://www.digital-detective.co.uk/">
```

If you wish to use ASP, then the following code inside an ASP page will perform the same function.

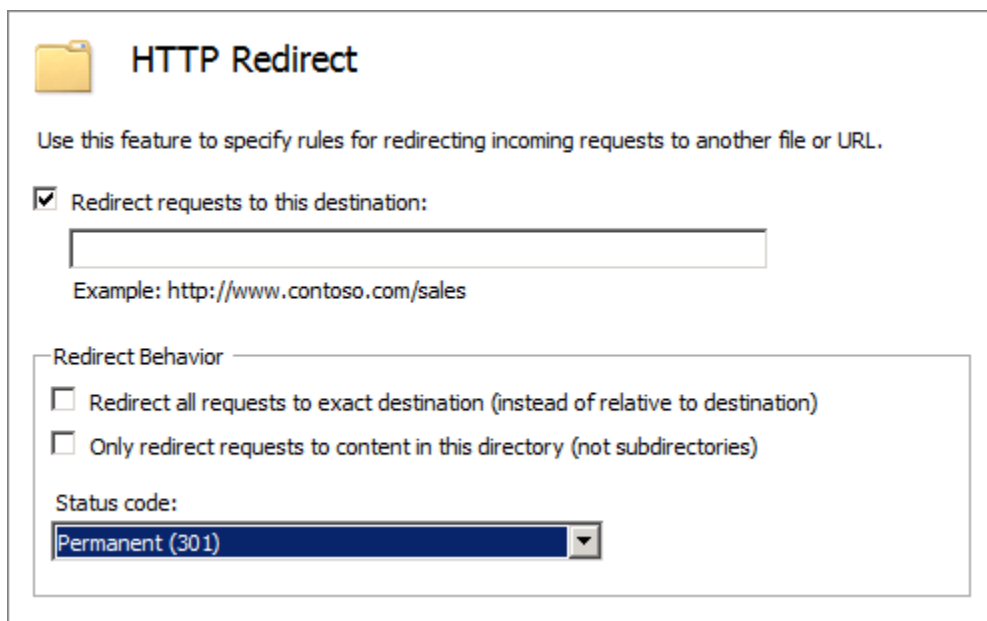
ASP Redirect

```
<% 'Begin buffering the HTML 'Note this MUST happen before the initial <HTML> tag. Response.Buffer = True %>
<HTML>
    <BODY>
        <% Response.Clear Response.Redirect "http://www.digital-detective.co.uk/" %>
    </BODY>
</html>
```

Server Side Redirects

There are many ways to implement server-side redirects depending on the web server being used. One of the most common is to use the .htaccess file, which is supported by the Apache web server. .htaccess supports a directive called redirect. This directive transparently changes the URL to a new URL. However, this is not the usual method for implementing Server-Side redirects as not all hosts support this.

In IIS7, creating server side redirect is a simple task. Log on to the Internet Information Services (IIS) Manager and select the file or folder you wish to create a redirect for. Clicking on the HTTP Redirect button will launch the screen shown in Figure 3. This allows the webmaster to select the URL to be redirected to, and also which type of redirect is required.



HTTP Redirect

Use this feature to specify rules for redirecting incoming requests to another file or URL.

☒ Redirect requests to this destination:

Example: `http://www.contoso.com/sales`

Redirect Behavior

☐ Redirect all requests to exact destination (instead of relative to destination)

☐ Only redirect requests to content in this directory (not subdirectories)

Status code:

Permanent (301)

Figure 3

This information is then saved to a `web.config` file in the folder where the redirect is to be launched from.

Redirects in Microsoft Internet Explorer

Embedded within the `CACHE INDEX.DAT` file are numerous Internet records that have `REDR` as the record header. This header is a `REDIRECT` entry and is evidence of a `SERVER-SIDE` redirect. Client Side redirects are **NOT** recorded within the `INDEX.DAT` files as `REDR` records. The `REDR` entry is a URL that has been visited and the server has responded with an `HTTP 300` response which tells the browser the page is in a different location. This entry reflects the URL which caused the redirection. the entries are marked as `Type: Redirect` with the entry shown below.

 redirect

In previous versions of NetAnalysis, we were not able to show where the user was redirected to. Following research and testing, we identified a methodology for resolving redirect entries. However, it is only possible to show the resulting redirected URL if the data still exists within the `INDEX.DAT` record. There is a marker to indicate whether the entry is live or deleted. It is also only possible to show resulting redirect entries from live `INDEX.DAT` files. It is not possible to do this with the data recovered by HstEx. This is a significant development in the analysis of Internet browser artefacts as previously, the investigator did not have this information. At the time of writing this Article, NetAnalysis v1.50+ is the only software available to extract this important evidence. In addition to identifying the URL where the user was redirected to, it is also possible to identify the date/time this action occurred. This also was not previously possible. The status column informs the examiner whether the redirect entry is intact or not. If it is intact and from a live `INDEX.DAT` file, there will be date/time information. As mentioned previously, this data is not available for overwritten or deleted redirect entries.

NetAnalysis v1.51 - Forensic Internet History Analysis - [Cache]

File Filter Searching Tools Bookmarks Reports Audit View Column Help

GMT Standard Time [UTC +0000]

0001 http://windows.microsoft.com/en-GB/windows/home 1

Type	Last Visited [Local]	URL	Redirect URL	IE Type	Status
redirect	21/04/2010 09:55:02 Wed	http://bit.ly/agtZtx	http://www.guardian.co.uk/books/2010/apr/19/penguin-cook-book	REDR	Redirected to:
redirect	23/04/2010 09:16:27 Fri	http://windows.microsoft.com/en-GB/...	http://www.microsoft.com/uk/windows/ 2	REDR	Redirected to:

www.digital-detective.co.uk Filter Cache P:\Temporary Internet Files\Content.IE5\index.dat FO: 80384 URL Records: 2

Figure 4

Figure 4 shows NetAnalysis and two Server Side Redirect entries. In this case, item number one is the original URL which caused the server to respond with a server side redirect HTTP response. This is the standard URL record which is shown in the URL column. In addition, as this is a REDR record which is intact, NetAnalysis has the Redirect URL (item number two) in the corresponding column. The Type column reflects the fact it is a redirect entry, as does the IE Type column (Internet Explorer Record Type). As this Redirect entry is intact, the Last Visited time stamp can be extracted.