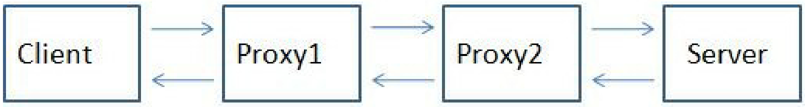
X-Forwarded-For Log Filter for Windows Servers

Joe Pruitt, 2009-19-08

For those that don't know what X-Forwarded-For is, then you might as well close your browser because this post likely will mean nothing to you…

A Little Background

Now, if you are still reading this, then you likely are having issues with determining the origin client connections to your web servers. When web requests are passed through proxies, load balancers, application delivery controllers, etc, the client no longer has a direct connection with the destination server and all traffic looks like it's coming from the last server in the chain. In the following diagram, Proxy2 is the last hop in the chain before the request hits the destination server. Relying on connection information alone, the server thinks that all connections come from Proxy2, not from the Client that initiated the connection.



The only one in the chain here who knows who the client really is (as determined by it's client IP Address, is Proxy1. The problem is that application owners rely on source client information for many reasons ranging from analyzing client demographics to targeting Denial of Service attacks.

That's where the X-Forwarded-For header comes in. It is non-RFC standard HTTP request header that is used for identifying the originating IP address of a client connecting to a web server through a proxy. The format of the header is:

> X-Forwarded-For: client, proxy1, proxy, …

X-Forwarded-For header logging is supported in Apache (with mod_proxy) but Microsoft IIS does not have a direct way to support the translation of the X-Forwarded-For value into the client ip (c-ip) header value used in its webserver logging.

Back in September, 2005 I wrote an ISAPI filter that can be installed within IIS to perform this transition. This was primarily for F5 customers but I figured that I might as well release it into the wild as others would find value out of it.

Recently folks have asked for 64 bit versions (especially with the release of Windows 2008 Server). This gave me the opportunity to brush up on my C skills. In addition to building targets for 64 bit windows, I went ahead and added a few new features that have been asked for.

Proxy Chain Support

The original implementation did not correctly parse the "client, proxy1, proxy2,…" format and assumed that there was a single IP address following the X-Forwarded-For header. I've added code to tokenize the values and strip out all but the first token in the comma delimited chain for inclusion in the logs.

Header Name Override

Others have asked to be able to change the header name that the filter looked for from "X-Forwarded-For" to some customized value. In some cases they were using the X-Forwarded-For header for another reason and wanted to use iRules to create a new header that was to be used in the logs. I implemented this by adding a configuration file option for the filter. The filter will look for a file named F5XForwardedFor.ini in the same directory as the filter with the following format:

```
[SETTINGS]
HEADER=Alternate-Header-Name
```

The value of "Alternate-Header-Name" can be changed to whatever header you would like to use.

Download

I've updated the original distribution file so that folks hitting my previous blog post would get the updates. The following zip file includes 32 and 64 bit release versions of the F5XForwardedFor.dll that you can install under IIS6 or IIS7.

Installation

Follow these steps to install the filter.

1. Download and unzip the F5XForwardedFor.zip distribution.
2. Copy the F5XForwardedFor.dll file from the x86\Release or x64\Release directory (depending on your platform) into a target directory on your system. Let's say C:\ISAPIFilters.
3. Ensure that the containing directory and the F5XForwardedFor.dll file have read permissions by the IIS process. It's easiest to just give full read access to everyone.
4. Open the IIS Admin utility and navigate to the web server you would like to apply it to.
5. For IIS6, Right click on your web server and select Properties. Then select the "ISAPI Filters" tab. From there click the "Add" button and enter "F5XForwardedFor" for the Name and the path to the file "c:\ISAPIFilters\F5XForwardedFor.dll" to the Executable field and click OK enough times to exit the property dialogs. At this point the filter should be working for you. You can go back into the property dialog to determine whether the filter is active or an error occurred.
6. For II7, you'll want to select your website and then double click on the "ISAPI Filters" icon that shows up in the Features View. In the Actions Pane on the right select the "Add" link and enter "F5XForwardedFor" for the name and "C:\ISAPIFilters\F5XForwardedFor.dll" for the Executable. Click OK and you are set to go.

I'd love to hear feedback on this and if there are any other feature request, I'm wide open to suggestions. The source code is included in the download distribution so if you make any changes yourself, let me know!

Good luck and happy filtering!

-Joe

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com