

Host Files and the Resolver Cache

Posted At : February 24, 2011 4:24 PM | Posted By : Mark Kruger

Related Categories: Hosting and Networking

This is something I assume everyone knows sometimes - but it's actually a networking thing. Surprisingly, quite a few very bright developers get that deer-in-the-headlights look when dealing with networking issues. The question is, "How do I test a domain without making it live." This is an important question. For example you might have code that is domain specific. That happens on those sites with a "shared codebase" or with Ajax or whatever. In addition, you might be developing on your local machine (many - perhaps most folks do) and want to set it up so that the domain `www.xyz.com` points to your local machine. You might have a site about to "go live" and you want to thoroughly test it prior to changing DNS. So there are many reasons you may need to do this. Here's a quick tutorial on how to use the hosts file to make this happen. Note - the examples are for Windows, but MAC and Linux also have hosts files so the principle still applies.

Hosts File

The *hosts* file is a file containing host and IP pairs. The system checks the HOSTS file to see if there's an entry for a particular domain. If it finds an entry, that IP is used for the domain. Now there is such a thing as "order of precedence" in domain resolution. The *default behavior* is for the IP stack to check the *hosts file first*. You can change this behavior and I've seen a couple of OEM custom installs that set it up so that DNS is used first (presumably because altering the hosts file is common hijack hack). But most of you won't have that problem.

On a windows box the file is found at `%systemroot%/system32/drivers/etc/hosts`. Note - there is no extension on the hosts file (it's not `hosts.txt`... it's just "hosts"). I always open it in notepad. The syntax of the hosts file is simple:

```
127.0.0.1      localhost
::1           localhost
127.0.0.1      sample.coldfusionmuse.com
10.10.10.1     t1.cfwebtools.com
#10.10.10.12   t2.cfwebtools.com
10.10.10.13    t3.cfwebtools.com
```

Pretty easy right? It's an IP address followed by a string. The string can be single name (like "serverA") or a fully qualified host/domain like `www.coldfusionmuse.com`. There are a couple items of note. The pound sign (#) or hash mark for my UK readers - is used to comment out a line. The line `::1 localhost` in the example above represents the local loopback address for the IPv6 stack. That's the new version of IP with 128bits instead of 32 - which means all of George Forman's children can have their own IP (plus every molecule on the planet). IPv6 isn't really in use in very many places as of yet.

The DNS Resolver Cache

Give it a try. Add this entry to your hosts file:

```
209.34.196.116 www.google.com
```

Then open a browser and navigate to `www.google.com`. Did it work? Probably not. To understand why you need to go to the command line (can I get a whoot from the Linux readers). Open a command line and issue this command:

```
C:\>ipconfig /displaydns
```

This will display *cached entries* for domains you have previously resolved. This is unimaginatively called the "resolver cache". It will also contain parsed entries from the hosts file. Take a close look at the entries returned by `ipconfig /flushdns`. It will contain a lot of information like time to live and reverse domain lookup entries. You might see some domains in there you don't recall visiting - as in "Hmmm... when was I ever on 'beatMeLikeARentedMule.com'?" Remember that a given web page might require resources from a dozen domains. HTML email calls for domain resources as well. And of course all the adware you are infested with keeps DNS pretty busy too - so it's likely your computer is enjoying the riches of the internet in ways you don't expect.

What you may *not* see is an entry matching `www.google.com` to `209.34.196.116`. Once the cache has a record for a domain (and Google is as ubiquitous as they come) it keeps it for the required "time to live". You can, however, flush the cache and start over. Try this command:

```
C:\>ipconfig /flushdns
```

After running a flush the IP stack re-parses the hosts file into the cache for you automatically. Now if you try `ipconfig /displaydns` you will see an entry like this:

```
www.google.com
-----
Record Name . . . . . : www.google.com
Record Type . . . . . : 1
Time To Live . . . . . : 0
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . : 209.34.196.116
```

Of course you can cut the chatter and simply ping the domains you are working with too. A combination of ping and `/flushdns` will get you to the point where you are resolving a domain to the IP you desire. In our example, if you open `www.google.com` in a browser you *should* see `coldfusionmuse.com` instead. Viola! Note - you would see whatever site is listening "by default" on that IP address. In this case, the IP I specified has a rule to serve the content of `www.coldfusionmuse.com` when it doesn't find a matching host header. Otherwise you might see some other site content you did not expect - whatever the IP serves up by default. So if you are doing this on your local machine you will need to know how to configure your web server to respond to the appropriate host header.

One More Tip

The most annoying thing about the host file is that it's buried deep and has no file extension. So here's what I do on my various desktops, laptops and servers.

- Navigate to `%systemroot%/system32/drivers/etc/hosts` using windows explorer.
- Drag the HOSTS file using the right mouse button to your desktop and choose "Crate shortcut here".
- Right click on the shortcut and choose properties.
- Change the *target* from `C:\WINDOWS\system32\drivers\etc\hosts` to `notepad.exe "C:\WINDOWS\system32\drivers\etc\hosts"` and click OK.

Your shortcut should change to the notepad icon. Clicking on it will bring up the HOSTS file for editing. Note, by default the hosts file is read only so you will have to reset that attribute (attrib or checkbox in the properties of the file) to make it editable. Hope this helps iron out the host file and resolver cache for you. Happy coding Muse readers!