

Forum:[Hack Forums](#)

Thread: [Prevent host file redirection](#)

Compoze - October 17, 2015, 5:04 am

Is there any way to prevent someone from redirecting requests, made by a program, using the host file?

I'm sure this thread fits into some of the other sections too but I'm just posting it here since it's a C# program.

Thanks in advance.

ubbelol - October 17, 2015, 5:25 am

If you're talking about denying other applications access to the hosts file then not really. Not easily at least.

Odus - October 17, 2015, 5:29 am

Assuming you do, how do you protect it when your application is closed? There's nothing to stop the user going back to it. Even if you set attributes to read only, they can still reverse it.

Compoze - October 17, 2015, 5:36 am

What I meant is I just want to validate that the response I get is from my webserver and has not tampered with or has been redirected to a local webserver by for example changing the host file. It doesn't matter what they do with the host file when my application is closed, as long as it's not used to modify requests while my program is open.

Cecilia - October 17, 2015, 5:38 am

Check whether the SSL certificate is the same?

I'm not absolutely sure.

You can also just check whether the hostname redirects to localhost.

N3w8i3H4ck3r - October 17, 2015, 1:19 pm

Just open cmd and ping the site, if the ip that is being returned is sketchy or local then you know stuff is being redirected.

Can also do "tracert" on the ip.

6e756c6c - October 17, 2015, 1:30 pm

Check out the DnsQuery API function and the DNS_QUERY_NO_HOSTS_FILE + DNS_QUERY_BYPASS_CACHE flags. Something to keep in mind though, someone could probably just spoof your DNS server, or one step further they could intercept the TCP packets to your server. In which case you'll need to use SSL (you'll need to verify the certificates) or you need handle your own key exchange and encryption at the application layer (requires asymmetric encryption, obviously) - but this can be bypassed if they patch your application.

My advice? Unless this application requires extreme security and you're expecting 1,000+ users don't exhaust the resources trying to secure it against anyone but the least experienced "script kiddies". Otherwise, it's going to get cracked, patched, and bypassed anyways but you'll have wasted countless development hours on it that could have been better spent elsewhere.

