Review: Foundeo Web Application Firewall

Posted At : April 17, 2009 7:44 PM | Posted By : Mark Kruger
Related Categories: ColdFusion, Product Reviews

Coldfusion Guru and all around super geek **Pete Freitag** was nice enough to let me check out his new "Web Application Firewall" (see **this link** for more info). This product serves as a Coldfusion based security filter for all requests coming to an application. I was impressed with the approach Pete takes. After the Init the code ran smoothly and did not appreciably increase my page load time (always a concern when you are "wrapping" your application in something). If you use Pete's system you will pretty much guarantee that your site will be protected from a high percentage of known attacks. Overall I give the application an A for innovation and organization, a B+ for installation and a C- on the price. If you want to know more read on.

A for Innovation

What's not to like. After you set it up you simply add a few lines of code to the top of your Application.cfm file or to the onRequest method of your Application.CFC file and your code is protected - nifty. The code has some protections you might not think of as well.

- **Repeat Offender** - The ability to lock out an IP after X number of attempts at the same thing. This is useful for blocking brute force attempts at hacking a password page (for example).
- **IP filter** - Put any IPs on the naughty list..
- **ID Blocker** - This is ingenious and I would not have thought of it. This filter blocks people from using a variable name ending in "ID" to send things like lists or complex strings. After installing the firewall into my application, this filter actually blocked several of my form requests. My application allowed a searcher to send in a list of IDS for things like status or users. So I had to disable this one to allow my application to continue to function. Still, I can see its usefulness.
- **SQL Injection Filter** - Very important... perhaps the most important filter.
- **Session High Jacking Filter** - I know of no other product that tries this from with the application code itself. This filter looks for changes in the session. I suspect that there might be some unintended consequences to this filter in high load environments.

There are a several other filters but you'll have to buy it to know them all :).

The firewall also has protection levels and log levels so you can adjust how picky you want to be. The logger uses cflog to write logs to the /cfusion8/logs directory (I would have preferred to be able to specify the directory). The log information is generally helpful.

Installation (B+)

For a Coldfusion Developer the install was a snap. It consisted of creating a /firewall folder off of the root of the application, copying and modifying a "configurator" file and adding a snippet like this one to the Application file.

```
<cfif NOT IsDefined("application.firewall")
                          OR isDefined('url.refresh')>
```

```
    <cfinvoke
            component="firewall.components.firewall"
            method="init"
            returnvariable="application.firewall">
        <!--- pass the name of the file as the argument --->
        <cfinvokeargument
            name="configurator"
                value="myConfiguratorFile">
    </cfinvoke>

</cfif>
```

When the code is run the filters are all loaded up and the firewall starts protecting your application right away.

It's not all a bed of roses however. The configurator file was not an XML file or an INI file (as I expected) but an actual CFC. To enable or disable a filter I was commenting it out or altering the values of CFSET statements. I would have expected a settings file of some kind that was more CF agnostic.

To ease the pain Pete has included a set of sample "configurators" that allow you to set your site up for "strict" or "basic" or even "log only". I can imagine using the "log only" version as a troubleshooting tool on a server with lots of traffic that I suspect of being attacked. It would produce a nice bucket full of information to help me along.

Extending the Firewall

You could make use of the filtering to make your application "smarter". Each filter is created using the same approach. For example here is how the IP blocker works:

```
<cfset ipFilter = application.firewall.newFilterInstance("SimpleIPBlockingFilter")>
<cfset ipFilter.blockIP("10.11.12.13")>
<cfset application.firewall.addFilter( filter )>
```

Imagine code within the application that handles IP lockout based on logic determined by the programmer and uses the application object to add an ip to the filter. A clever programmer could further extend the application by adding his own filters.

Optimistic Security

Although it is called a firewall, the security model here is optimistic rather than pessimistic - that is to say this product *lets everything in* by default and blocks bad stuff. A firewall in the traditional sense (a routing device on the network perimeter) is supposed to *block everything* by default and then *allow* the good stuff. That is the reverse approach in theory. In practice the best firewalls do more than just allow traffic through specific ports. They also inspect the packets to determine if something bad is sneaking in as if it were good. The optimistic security approach is more like a spam filter. It can protect you against *everything it recognizes as spam*.

In the same way this filter can protect you against *everything it can recognize as bad*, but that is not the same as blocking everything and only allowing requests that are recognizably good. It's an important distinction. An application like this can be extremely effective at securing your application from practically all "known" threats. That should not make you stop worrying. Instead, it should make you stop worrying about the threats you know about so you can concentrate on the ones you don't know about.

Cost (C-)

I applaud Pete for aiming high. His product is unique and it may indeed be worth $500 an application (or $1300 per server or $9k for a company) - at least in altruistic sense. However, for that amount of money it will need more polish and it will have to be more that a set of CF scripts. It will need a control panel for deployment, settings and real time filtering (like banning an IP address for a particular application for example). It might need to have a log viewer with some analysis. It could benefit from a tie in with SeeFusion or FusionReactor etc. I wish him well, but I suspect he'll get few hits at that price level. Still it never hurts to aim high, you can always bring the price down. I think it is worthy of a look for some folks, but I believe the price will be a sticking point for most of them - even deep pocketed corporations.

Now, having said that, remember that I have do not necessarily have the greatest take on the topic. If the product takes off and everyone is standing in line to buy it I will be the first one applauding ('cause I like Pete and I have always appreciated his stuff). I'm simply sharing my gut feeling in the interest of an honest reviewer.