IIS Vulnerability Steals Payment Information (By Wil Genovese - CFG)

Posted At : March 6, 2014 2:35 PM | Posted By : Mark Kruger
Related Categories: Coldfusion Security, ColdFusion

Super guru Wil Genovese (**Trunkful.com**) is back to describe an IIS vulnerability that was inserted using a long-known (and patched) CF vulnerability. The Muse will make 2 points. First, if you are hit with this one call us! We will gladly put our shoulder to the wheel and help you dig out. Second, don't forget to patch your servers and keep up on the latest security news. No matter what your chosen platform you need to be vigilant and attentive. Take it away Wil.

First let me point out that the vulnerability that was found has a patch that has been available since January of 2013. So as the Muse said, patch your servers! I first read about this attack in a PC World article titled, **PCWorld - Attackers exploited ColdFusion vulnerability to install Microsoft IIS malware**. I spent hours reading all the linked websites and blog posts by the security researcher that discovered the IIS Malware (see this **Trustwave** post) trying in vain to learn the name of said DLL that gets installed, where it gets installed and how to detect the file(s). The few details I found were not completely useful. While I learned the behavior of the malware I never learned how to find the offending DLL or even the file name. I*did* discover that no existing anti-malware or anti-virus software would detect this rogue DLL. I repeated my futile search every few weeks to see if anything new was being reported.

Since knowing how to locate and expunge such things is part of my job I needed a way to find it, but how? I could search any of the servers at CF Webtools until the cows come home, but if none of them have been hit with this malware I will never find it. What I needed was a server that had been exploited to examine. Over the past year with the slightly larger than usual number of security holes discovered in ColdFusion we've had a few new clients come to us for help in patching and repairing servers. None of the IIS modules on those servers stood out to me as 'unusual', but I wasn't looking directly for this. Finally we had a company come to us for help with a breach.

When describing the issues they were having it sounded familiar. If you haven't read the link above let me describe what happens. Using an unpatched ColdFusion server an attacker uses the exploit describe by Adobe as CVE-2013-0629. Adobe actually fixed this vulnerability in January 2013 with **this patch**. The vulnerability allows the attacker to install a **Web Shell**. A Web Shell is a simple web application that allows the attacker to have full file system access and more. Using the Web Shell the attacker uploads and executes an installer that registers the DLL and a adds a Global Module to IIS. Although the researchers have dubbed the module "ISN", this is not the actual name of the DLL. In fact according to the researchers it could be named anything.

> "the researchers revealed that ISN is being installed on the compromised IIS servers by exploiting a remote authentication bypass vulnerability in Adobe ColdFusion, a Web application platform." - PC World

The way this rogue IIS module works is to monitor POST requests for specific file names which are added as patterns at the time of install. For example the installer could have added "processcc.cfm" as an argument when the dll was installed. When it sees a post to that file it captures the data from the file into a log file. Essentially it's sniffing the POST calls to your payment processing CFML file. It can function even with SSL enabled, because it can access data at the server level. It captures the data*after* the

SSL POST is decrypted by the server. The attacker is able to retrieve the data stored in the log file because the rogue IIS module will respond to 'special' URL parameters.

In this case the client was telling me that no mater what they or their hosting company did to patch or lockdown the server, credit card data was still being stolen. Their payment processor was working with them to help resolve the issue and eventually all payment processing was moved off of the server to the payment processor. This stopped the data loss, but still no one knew how it was happening.

I reviewed the notes from the hosting provider about the items they found in the past and I saw the usual suspects for exploits we know about. Namely the "h.cfm" and "i.cfm" files via the Scheduled Tasks flaw. A little more digging in the file system and I found a "Web Shell". In the IIS logs I could see the date and time that the scheduler API had been accessed. Over the course of working on the server I was actually tripping over the offending DLL, but I didn't recognize it at first as anything that would be suspicious. It looked like something that was supposed to be there. The best attackers can hide in plain sight.

How did I find it? I was trying to remove the IIS connectors for ColdFusion 8. Meaning I was removing the jrun_iis6.dll and other related settings from IIS. I had to do it manually because wsconfig would not run. Yet whenever I tried to access the website (even just an html page) one JRun setting kept being reinserted into the config files. This happened several times. I tried removing this from the applicationHost.config file (the main IIS config file), but after the next http request to IIS it was back. This pesky **jrur jrun_iis.dll** kept returning no matter what I tried. I must have tripped over it 4 or 5 times before I stopped and decided to look closely at it. The DLL was located in "c:\ColdFusion8\runtime\lib\wsconfig\jrun_iis.dll," and next to it in the same folder was a config file named "jrun_iis.cfg". When I opened the config file I saw the names of two CFML files that looked suspiciously like they were related to payments. I confirmed with the client that these two files were indeed his old payment processing files. There it is, the "Unnamed DLL", sitting there masquerading as a JRun connector DLL! So I finally knew the name of the rogue IIS module - at least the name it used on *this* **this**server. Obviously by this time we recognized how significant the exploit was. We advised the client to ditch the server immediately and move to a new VPS intance (freshly patched and installed from scratch).

Just the Facts Ma'am

The critical thing to remember is that this is an exploit on IIS that used hole in ColdFusion as the path to installation. A similar hole in any web application in any language could allow the same IIS module to be installed. Here are the details I developed regarding this IIS Module. The module was mimicking a JRun connector. The module monitors HTTP POSTS and not ColdFusion itself. A cfg file contains the details of the actual scripts (file names) in the post request that it was tasked to monitor. In the IIS applicationHost.config file, under Global Modules I could see a JRun Module.

```
C:\windows\system32\
<globalModules>
...
<add name="jrun_iis" image="c:\ColdFusion8\runtime\lib\wsconfig\jrun_iis.dll" />
...
</globalModules>
```

The module install also added these web.config files to each website on the server

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
<system.webServer>
<modules>
<add name="jrun_iis" />
</modules>
</system.webServer>

</configuration>
```

Even after removing the lines from the various config files and the application pool config files this module reappears - being reinstalled with any new request to any web resource. We did find that install mechanism but something is monitoring and reinstalling it. If I just removed the DLL then IIS fails to run the web site due to a missing DLL. This is both maddeningly unhelpful and fiendishly clever. It is also important to remember that according to the researchers at Trustwave the module can be named anything. In this case it was hiding among the other JRun DLL's and thus looked like JRun. On a different system it could look like something else. I think the key is to inspect the Global Modules in IIS to see if something is pretending to be what it's not supposed to be.

Finally, this is yet another warning to PATCH YOUR SERVERS! It's is also a reminder to follow the ColdFusion Lockdown Guides from Adobe. I highly recommend that anyone that was hit by the above "h.cfm" exploit via the hole that was in the ColdFusion Scheduler Tasks API to go back and inspect your servers.