Detecting HAFNIUM Exchange Server Zero-Day Activity in Splunk

splunk.com/en_us/blog/security/detecting-hafnium-exchange-server-zero-day-activity-in-splunk.html

March 3, 2021

By Ryan Kovar March 03, 2021

If you want just to see how to find HAFNIUM Exchange Zero-Day Activity, skip down to the "detections" sections. Otherwise, read on for a quick breakdown of what happened, how to detect it, and MITRE ATT&CK mappings.

Introduction to HAFNIUM and the Exchange Zero-Day Activity

On Tuesday, March 2, 2021, Microsoft released a set of security patches for its mail server, Microsoft Exchange. These patches respond to a group of vulnerabilities known to impact Exchange 2013, 2016, and 2019. It is important to note that an Exchange 2010 security update has also been issued, though the CVEs do not reference that version as being vulnerable.

While the CVEs do not shed much light on the specifics of the vulnerabilities or exploits, the first vulnerability (CVE-2021-26855) has a remote network attack vector that allows the attacker, a group Microsoft named HAFNIUM, to authenticate as the Exchange server. Three additional vulnerabilities (CVE-2021-26857, CVE-2021-26858, and CVE-2021-27065) were also identified as part of this activity. When chained together along with CVE-2021-26855 for initial access, the attacker would have complete control over the Exchange server. This includes the ability to run code as SYSTEM and write to any path on the server.

A temporary mitigation for these vulnerabilities from external threats is restricting access to OWA, such as placing the OWA server behind a VPN to prevent external access. This does

not, however, prevent an internal attacker from exploiting the vulnerability. In short, patch as soon as possible.

What you need to know

You may be thinking, “another Tuesday filled with patches, just like any other month.” That may be true to some extent, but it is essential to point out based on Volexity’s blog that:

> ***“In all cases of RCE (remote code execution), Volexity has observed the attacker writing web shells (ASPX files) to disk and conducting further operations to dump credentials, add user accounts, steal copies of the Active Directory database (NTDS.DIT), and move laterally to other systems and environments.”***

I don’t know about you, but whenever I see an adversary stealing copies of my Active Directory (AD) database, that sends chills down my spine because, at that point, I am rebuilding my entire AD from scratch as part of my remediation effort. For more color,

stealing the AD database implies that the adversary will have domain administrator privilege, so this is important to investigate.

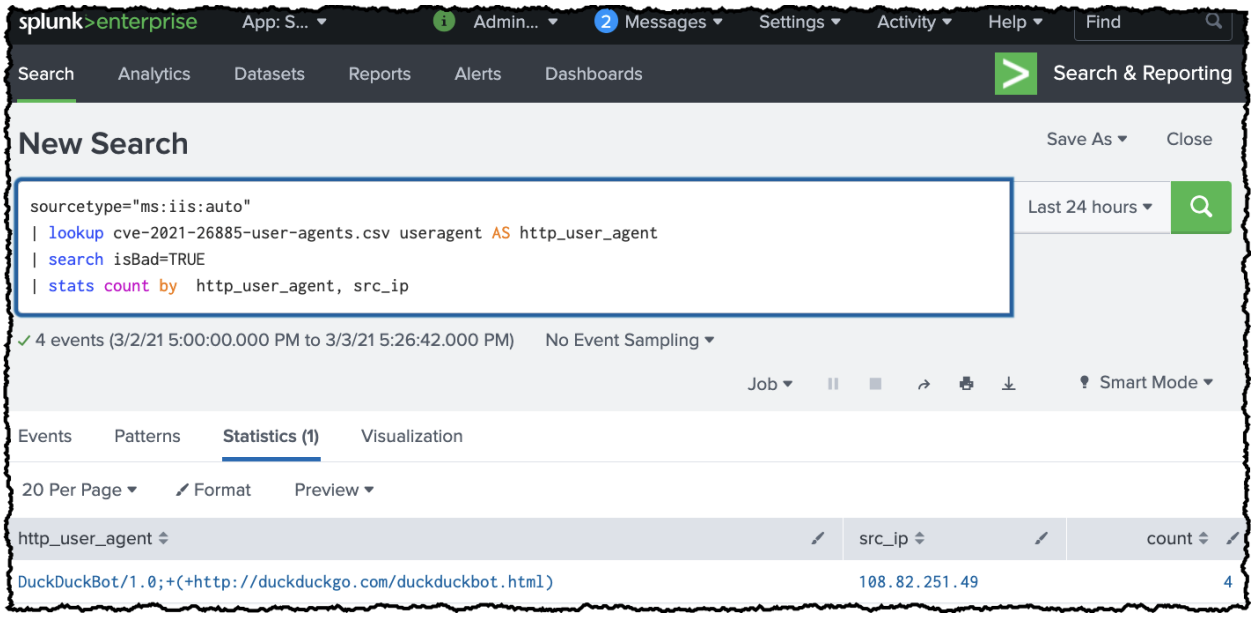
OWA! That hurts!

Some of these vulnerabilities are being exploited via Outlook Web Services (OWA), a commonly enabled feature of Exchange Server 2013, 2016, and 2019. Underlying OWA is Microsoft's venerable web server, Internet Information Services (IIS). It just so happens that elements of this attack can be detected by looking for the appropriate POST requests in IIS logs.

Wait just a moment! Splunk is super good at ingesting logs from all sources and looking for patterns in them! All we need to do is ensure that logs are being ingested from our OWA servers appropriately.

Our recipe for success is to use the Splunk Universal Forwarder and add in a little bit of the Splunk-supported Technical Add-On for Microsoft IIS. Next, add something like this to your inputs.conf file so that you can ingest all of the exciting logs in the C:\inetpub\logs\LogFiles directory in W3C format. This will let you search through the IIS access logs for unusual User-Agent string patterns known to be associated with this attack, as was mentioned earlier today by our friends at Red Canary. You'll also want to add a monitoring entry to capture log activity in C:\Program Files\Microsoft\Exchange Server\V15\Logging\HttpProxy.

Refer to the Volexity blog post for other interesting User-Agents seen both pre and post-exploit.



Yes, Yet Again, We Need Windows Events and Command Line Auditing

While you're under the hood of that Universal Forwarder, you might as well ensure you're collecting both Windows Security events of interest and process start events, as both of those are important for certain HAFNIUM detections. For this, we recommend you use the Technical Add-On for Windows and also collect event 4688 (with command line arguments) from the Security event log, OR you can always use Microsoft Sysmon and collect event 1. These process start events can be searched for anomalous execution of the Exchange UM service.

You can also configure auditing on your Exchange server UM process and then search for Windows 4663 events for suspicious FileCreated events (in this case, the web shells.) However, be careful here - the possibility to generate many thousands of 4663 events exists if you do not set up your auditing policies correctly!

Finally, collect the Application log from your Exchange servers, again using the Windows TA. This allows you to search for errors in the log containing specific patterns in the RenderedDescription field about the Exchange UM service's execution.

Detecting HAFNIUM and Exchange Zero-Day Activity in Splunk

Here we will give you some hot-off-the-press searches to help find some of the HAFNIUM badness derived from the Volexity and Microsoft blogs. If we have coverage for these searches in ESCU, we call them out further below in the MITRE ATT&CK section.

Please note that the actual remote code execution (RCE) proof of concept (POC) for these CVEs has not been publicly released. As such, the detections below are all derived from the original blogs from Microsoft and Volexity. When we have more information, we will publish updates!

Indicators of Compromise (IOCs)

Both Volexity and Microsoft published IOCs, including IP addresses of observed attackers, web shell hashes and filenames, and user-agents in their blog posts. We have converted these indicators into simple CSV format so that you may use them as lookup tables - they are posted here. But what's a lookup table, and how does it help with security detection in Splunk? Got you covered there, too.

Authentication Bypass Vulnerability

From Volexity's blog, we see that this exploit is bypassing authentication and then moving to access users' emails with minimal effort. The exploit requires an HTTP POST to files in a specific directory /owa/auth/Current/themes/resources/.

We can detect this behavior via the following search, which requires ingesting the IIS logs on the Exchange OWA server.

```
index=* sourcetype="ms:iis:auto" http_method=POST
uri_path="/owa/auth/Current/themes/resources/*"
| stats count by src_ip, http_user_agent, uri_path, http_method, uri_query
```

Nishang PowerShell framework

One of the tools used by HAFNIUM in this attack was using the Nishang PowerShell framework. One method would be to look for PowerShell messages where the Nishang commandlets called out in the Microsoft blog would be detected:

```
index="*" sourcetype="WinEventLog" source="WinEventLog:Security" EventCode=4104
Message="* Invoke-PowerShellTCP*"
```

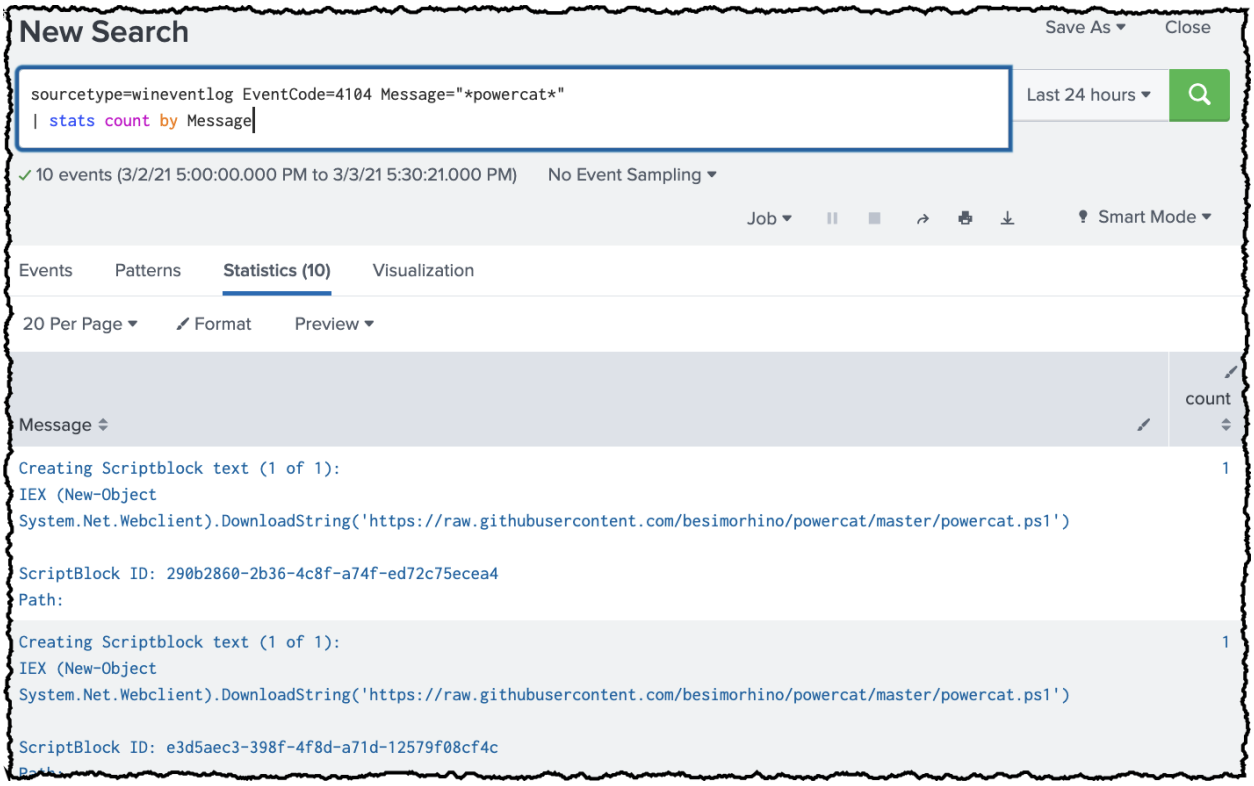
Another would be to use other would use event code 4688 and find some of the specific activity executed:

```
index="*" sourcetype="WinEventLog" source="WinEventLog:Security" EventCode=4688
Creator_Process_Name="powershell.exe" System.Net.Sockets.TCPClient
```

Powercat detection

The adversary used PowerShell to download and execute Powercat in memory. We've talked about PowerShell and IEX cradles for years at Splunk (like this blog and this .conf16 talk). One simple way to detect if you were affected by this activity is to make sure you are bringing in event code 4104 and check for "powercat":

```
index="*" sourcetype="WinEventLog" source="WinEventLog:Security" EventCode=4104
Message="*powercat*"
```



Exchange Unified Messaging Service Creating Executable Content

The Unified Messaging Service in Microsoft Exchange Server may be abused by HAFNIUM to create executable content in common server-side formats such as PHO, JSP, ASPX, etc. It is unusual and dangerous for a server-side component like the Unified Messaging Service to create new executable content. Such content, if subsequently executed, could be used for remote code execution. This search inspects Event ID 4663 file system audit logs for evidence that the Unified Messaging service has been misused to create new executable content on the server.

```
index=* sourcetype=WinEventLog source="WinEventLog:Security" EventCode=4663
Object_Type="File" (Object_Name="*.php" OR Object_Name="*.jsp" OR Object_Name="*.js"
OR Object_Name="*.aspx" OR Object_Name="*.asmx" OR Object_Name="*.cfm" OR
Object_Name="*.shtml") (Process_Name="*umworkerprocess.exe*" OR
Process_Name="*UMService.exe*")
```

Exchange Unified Messaging Service Creating Child Process

The Unified Messaging Service in Microsoft Exchange Server may be abused by HAFNIUM to launch child processes. Watching for unusual parent processes is a well-established detection technique that we can adapt and apply in this situation. This Splunk search takes advantage of Windows Event ID 4688, also referred to as Process Creation events. When the parent process is related to Exchange Unified Messaging, the process may be suspicious. This search employs a NOT clause to help reduce noise.

```
index=* sourcetype="WinEventLog" source="WinEventLog:Security" EventCode=4688
(Creator_Process_Name="*umworkerprocess.exe*" OR
Creator_Process_Name="*UMService.exe*") NOT New_Process_Name="*UMWorkerProcess.exe*"
```

Finding Hacker Tools

Suspicious zip, rar, and 7z files that are created in C:\ProgramData\ may indicate possible data staging for exfiltration. The searches below for Sysmon and Windows Event logs, respectively, may assist in identifying these files.

```
index=* (sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" OR
source="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational") EventCode=11
(TargetFilename="C:\\ProgramData\\*.rar" OR TargetFilename="C:\\ProgramData\\*.zip"
OR TargetFilename="C:\\ProgramData\\*.7z")
```

OR

```
index=* (sourcetype="WinEventLog" source="WinEventLog:Security"  OR
sourcetype=XmlWinEventLog source="XmlWinEventLog:Security") EventCode=4663
AccessList="%%4417" (ObjectName="C:\\ProgramData\\*.rar" OR
ObjectName="C:\\ProgramData\\*.zip" OR ObjectName="C:\\ProgramData\\*.7z")
```

It has also been noted that the execution of 7-Zip to create archives as a means of staging data for exfiltration has been utilized as well. To determine if this specific file has been executed, the following searches using Sysmon and Windows Event logs, respectively, can be used as a starting point.

```
index=* (sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational" OR
source="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational") EventCode=1
CommandLine="C:\\ProgramData\\7z*"
```

OR

```
index=* (sourcetype="WinEventLog" source="WinEventLog:Security"  OR
sourcetype=XmlWinEventLog source="XmlWinEventLog:Security") EventCode=4688
process="C:\\ProgramData\\7z*"
```

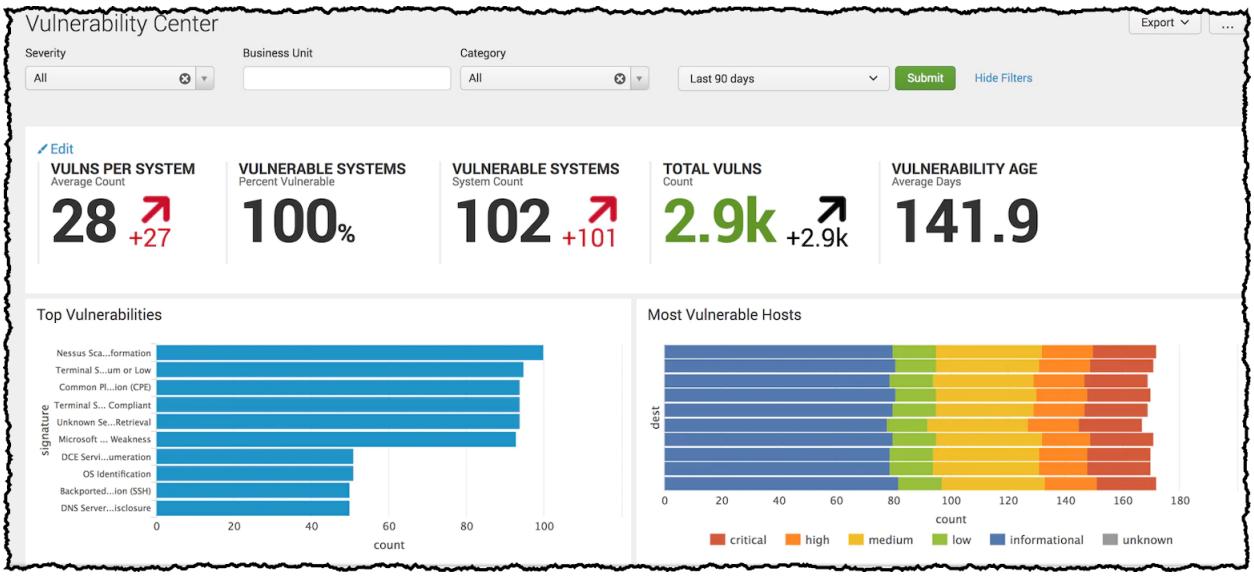
Web Shells

If you don’t have IIS logs, don’t fret, Wire data (like Splunk for Stream) or Zeek logs that capture web traffic can also serve as a way of gaining some visibility. Obviously, we need a “Hunting with Splunk” post on the subject! However, until then, check out this classic Youtube video from James Bower about how to hunt web shells with Splunk.

Splunk Enterprise Security and ESCU

Know thyself

While we have spent some time explaining that this attack is essential and effort needs to be put toward investigating this, it is also important to note that the basics are important. Basic asset management, hopefully via your asset and identity framework, will tell you where your Exchange servers reside. Running regular vulnerability scans that integrate into Splunk will display which Exchange servers are vulnerable and can help you prioritize your patching schedule and better focus your detection efforts.



Threat Intelligence Framework

If you are using Splunk Enterprise Security, the lookups of IOCs that are listed above can be ingested easily into the threat intelligence framework. Perhaps you aren't sure how to do that. No worries, we published some guidance and a how-to on integrating lists of IOC into the Enterprise Security threat intelligence framework.

Enterprise Security Content Updates (ESCU)

For folks using ESCU, our Security Research team will release a new Splunk Analytic Story called "HAFNIUM Group" on March 4th, 2021, containing detections for this threat. Saying that, check out the MITRE ATT&CK table below. If you have ESCU running today, you already have some great coverage!

MITRE ATT&CK

Reviewing the blog posts from Microsoft and Volexity, we mapped the adversary's activity to MITRE ATT&CK. Each tactic is then linked to Splunk Content to help you hunt for that information. Be aware; these searches are provided as a way to accelerate your hunting. We recommend you configure them via the Splunk Security Essentials App. You may need to modify them to work in your environment! Many of these searches are optimized for use with the tstats command.

Finally, as more information becomes available, we will update these searches if more ATT&CK TTPs become known.

ATT&CK Tactic	Title	HAFNIUM activity	Splunk Searches
T1003.001	OS Credential Dumping: LSASS Memory	Used Procdump to export LSASS	Dump LSASS via Procdump Dump of LSASS using comsvcs.dll
T1059.001	Command and Scripting Interpreter: PowerShell	Nishang PowerShell	Malicious PowerShell Process - Connect To Internet With Hidden Window, Malicious PowerShell Process - Execution Policy Bypass Attempt To Set Default PowerShell Execution Policy To Unrestricted or Bypass
T1114.001	Email Collection: Local Email Collection	PowerShell mailbox collection	Email files written outside of the email directory
T1136	Create Account	Add user accounts	Detect New Local Admin account
T1003.003	OS Credential Dumping: NTDS	Steal copies of the Active Directory database (NTDS.DIT)	Ntdsutil export ntds
T1021.002	Remote Services: SMB/Windows Admin Shares	Lateral movement	Detect PsExec With accepteula Flag

Here is a list of all the MITRE ATT&CK TTP's that we saw being used in this attack: T1003.001, T1005, T1027, T1046, T1059, T1059.001, T1070, T1071, T1074.002, T1083, T1110, T1190, T1505, T1560.001, T1589.002, T1590.002

Conclusion

We know that this is a significant vulnerability and that customers will want to patch as soon as possible and determine if they were affected by this attack in the past. If you haven't patched yet (we've all been there), hopefully, these searches will provide you the ability to have more visibility into your environment and any malicious activity that you might be experiencing. If they don't work perfectly, think of them as "SplunkSpiration" :-). As soon as we have more information, we will update this blog and, as we talked about earlier, be on the lookout for more detections from our threat research team that will be released through Enterprise Security Content Updates.

As always, security at Splunk is a family business. Credit to authors and collaborators:

Mick Baccio
James Brodsky
Shannon Davis
Michael Haag
Amy Heng
Jose Hernandez
Dave Herrald
Ryan Kovar
Marcus LaFerrera
John Stoner

Posted by

Ryan Kovar

NY. AZ. Navy. SOCA. KBMG. DARPA. Splunk.