ProxyShell vulnerabilities in Microsoft Exchange: What to do

news.sophos.com/en-us/2021/08/23/proxyshell-vulnerabilities-in-microsoft-exchange-what-to-do/

Greg Iddon

August 23, 2021

Last updated 2021-09-23 UTC 11.26

Overview

Threat actors are actively scanning and exploiting vulnerable Microsoft Exchange servers that have not applied security patches released earlier this year.

ProxyShell, the name given to a collection of vulnerabilities for Microsoft Exchange servers, enables an actor to bypass authentication and execute code as a privileged user.

ProxyShell comprises three separate vulnerabilities used as part of a single attack chain:

- **CVE-2021-34473**
  Pre-auth path confusion vulnerability to bypass access control
  Patched in KB5001779, released in April
- **CVE-2021-34523**
  Privilege elevation vulnerability in the Exchange PowerShell backend
  Patched in KB5001779, released in April
- **CVE-2021-31207**
  Post-auth remote code execution via arbitrary file write
  Patched in KB5003435, released in May

The vulnerabilities lie in the Microsoft Client Access Service (CAS) that typically runs on port 443 in IIS (Microsoft's web server). CAS is commonly exposed to the public internet to enable users to access their email via mobile devices and web browsers. This exposure has led to widespread exploitation by threat actors who are commonly deploying web shells to remotely execute arbitrary code on compromised devices, similar to that seen in the HAFNIUM attack.

What should you do?

Watch the video above as Mat Gangwer, head of the Sophos Managed Threat Response (MTR) team, shares details about the threat and offers advice about how to respond.

If you are using Microsoft Exchange server:

1. Backup Exchange IIS/Server logs and ensure you have applied the July 2021 security updates for Microsoft Exchange

   Patching only ensures that the vulnerability cannot be further exploited. If you have already been breached, the software patches do not address post-exploit behavior by a threat actor
2. (For non Sophos MTR customers) Identify and investigate your exposure windows for adversarial activity
   - Identify and delete web shells and malicious binaries
   - Review process activity for instances of `w3wp.exe`
   - Identify and remove any persistence established by an actor
3. Ensure endpoint protection is deployed on all endpoints and servers. Verify that all protections have been enabled and your exclusions are kept to a minimum

Sophos detections

Sophos customers are protected by multiple detections for the exploitation of these vulnerabilities. They can be used by threat hunters to perform searches in their own environments. Detections include:

- Troj/ASPDoor-Y (detects malicious PST files)
- Troj/ASPDoor-AF (detects malicious PST files)
- Troj/Agent-BHPF
- Troj/Agent-BHQD (detects the binary component of LockFile ransomware)
- Troj/WebShel-M
- Troj/KillAV-IT
- App/HamaKaze-A
- App/HamaKaze-B
- CXmal/WebAgnt-A (detects malicious PST files in the context of customers' environments)

SophosLabs has also published IPS signatures:

CVE	Sophos XG/ Sophos Firewall	EIPS	SG UTM
CVE-2021-34473	2305889, 2305807, 2305979	2305807	57906, 57907, 57908, 57909

In addition, on August 24th, SophosLabs released a new, more generic signature 2305979 to detect attempted vulnerability exploit in Microsoft Exchange server.

LockFile is a new ransomware family that appears to exploit the ProxyShell vulnerabilities to breach targets with unpatched, on premises Microsoft Exchange servers. SophosLabs has released additional behavior-based protection for LockFile provided by the Mem/LockFile-A detection for Windows devices running Sophos endpoint and server protection managed through Sophos Central.

Determining impact with Sophos XDR

1. Investigate exposure

Verifying current Microsoft Exchange version

To determine whether you are running an unpatched version of Exchange or not, the below XDR query for live Windows devices will produce a table of Exchange servers, their current version, and guidance whether they need patching or not.

The version numbers identified in the below query were gathered from this Microsoft article.

```
SELECT DISTINCT
  'Check Exchange Version to confirm Patch. Manually verify build number from MS
documentation./' Note,
  CASE product_version
    WHEN '15.2.922.13' THEN 'Exchange 2019 CU10 Jul21 patched against ProxyShell'
    WHEN '15.2.922.7' THEN 'Exchange 2019 CU10 patched against ProxyShell. Recommend
also updating with recent July Patch.'
    WHEN '15.2.858.15' THEN 'Exchange 2019 CU9 Jul21 patched against ProxyShell'
    WHEN '15.2.858.12' THEN 'Exchange 2019 CU9 May21 patched against ProxyShell.
Recommend also updating with recent July Patch.'
    WHEN '15.1.2308.14' THEN 'Exchange 2016 CU21 Jul21 patched against ProxyShell'
    WHEN '15.1.2308.8' THEN 'Exchange 2016 CU21 patched against ProxyShell. Recommend
also updating with recent July Patch.'
    WHEN '15.1.2242.12' THEN 'Exchange 2016 CU21 Jul21 patched against ProxyShell.'
    WHEN '15.1.2242.10' THEN 'Exchange 2016 CU20 May21  patched against ProxyShell.
Recommend also updating with recent July Patch.'
    WHEN '15.1.2176.14' THEN 'Exchange 2016 CU19 May21  patched against ProxyShell.
Recommend also updating with recent July Patch.'
    WHEN '15.0.1497.23' THEN 'Exchange 2013 CU23 Jul21 patched against ProxyShell.'
    WHEN '15.0.1497.18' THEN 'Exchange 2013 CU23 May21 patched against ProxyShell.
Recommend also updating with recent July Patch.'
    ELSE 'NOT PATCHED'
  END Result,
  'Product_Version: ' || Product_version Evidence
FROM file
WHERE path =
  ((
    SELECT data FROM registry
    WHERE key = 'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ExchangeServer\v15\Setup' AND
path =
'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ExchangeServer\v15\Setup\MsiInstallPath'
  )||'bin\Microsoft.Exchange.RpcClientAccess.Service.exe')
```

Analyze IIS logs for autodiscover.json abuse

As these vulnerabilities lie in the Exchange Client Access Service (CAS) which runs over IIS (web server), reviewing the IIS logs will reveal attempted and successful exploitation of the ProxyShell vulnerabilities. HTTP requests inbound to the IIS server will be detailed including the request type and path.

By default, IIS logs are written to `C:\inetpub\logs\LogFiles\`

A common artifact seen in these logs for abuse of CVE-2021-34473 is the presence of `&Email=autodiscover/autodiscover.json` in the request path to confuse the Exchange proxy to erroneously strip the wrong part from the URL.

E.g. `GET /autodiscover/autodiscover.json @evilcorp/ews/exchange.asmx?&Email=autodiscover/autodiscover.json%3F@evil.corp`

The below XDR query for live Windows devices will query the IIS logs on disk for any lines that contain the string ‘autodiscover.json’.

Should you later identify web shells, this same query can be repurposed to query for the web shell file name to reveal requests made to the web shell – simply change ‘autodiscover.json’ to ‘webshell_name.aspx’. Please note that this query can be slow depending on the volume of logs it needs to parse.

```
SELECT grep.*
FROM file
CROSS JOIN grep ON (grep.path = file.path)
WHERE
file.path LIKE 'C:\inetpub\logs\LogFiles\W3SVC%\u_ex210[89]%'
AND grep.pattern = 'autodiscover.json'
```

Windows Events for New-MailboxExportRequest abuse

CVE-2021-31207 enables a threat actor to write files to disk by abusing a feature of the Exchange PowerShell backend, specifically the `New-MailboxExportRequest` cmdlet. This cmdlet enables an email to be written to disk, using a UNC path, that contains an arbitrary email attachment. This has been the primary method used to deliver a web shell to a compromised device.

Windows Event logs for MSExchange Management typically log usage of `New-MailboxExportRequest`. By reviewing these logs, the locations of web shells can be ascertained.

The below XDR query for live Windows devices will query the Windows Event logs from the past 14 days for any events that detail usage of this cmdlet and the parameters of the command (including file path).

```
SELECT *
FROM sophos_windows_events
WHERE source = 'MSExchange Management'
AND time > strftime('%s', 'now', '-14 days')
AND data LIKE '%MailboxExportRequest%'
```

2. Identify suspicious web shells and binaries

Adversaries exploiting these vulnerabilities are dropping web shells on to the compromised device through which they can issue additional commands such as downloading and executing malicious binaries (such as `.exe` or `.dll` files).

As these vulnerabilities lie in CAS which runs on IIS, adversarial activity will stem from a w3wp.exe process, a worker process for IIS.

Web shells on disk

The below XDR query for live Windows devices looks at directories where adversaries are dropping web shells which may still be present on disk. Review any unexpected or recently created `.aspx` files that are present in the output of the query.

E.g. `C:\inetpub\wwwroot\aspnet_client\654253568.aspx`

```
SELECT * FROM
file sf
LEFT JOIN hash sh
ON sf.path = sh.path
WHERE
sf.path LIKE 'C:\inetpub\wwwroot\aspnet_client\system_web\%.aspx'
OR sf.path LIKE 'C:\inetpub\wwwroot\aspnet_client\%.aspx'
OR sf.path LIKE 'C:\Program Files\Microsoft\Exchange
Server\V15\FrontEnd\HttpProxy\owa\auth\%.aspx'
OR sf.path LIKE 'C:\Program Files\Microsoft\Exchange
Server\V15\FrontEnd\HttpProxy\ecp\auth\%.aspx'
OR sf.path LIKE 'C:\Program Files\Microsoft\Exchange
Server\V15\FrontEnd\HttpProxy\owa\auth\current\%.aspx'
OR sf.path LIKE 'C:\Program Files\Microsoft\Exchange
Server\V15\FrontEnd\HttpProxy\owa\auth\current\themes\%.aspx'
OR sf.path LIKE 'C:\ProgramData\%.aspx'
OR sf.path LIKE 'C:\ProgramData\%\%.aspx'
```

With the results, you can pivot from the path column of a suspected web shell by clicking the `(…)` button and selecting “File access history” to query and identify what processes have interacted with the file and which process created the file. Instances of `w3wp.exe` should be investigated to reveal further actions the adversary may have taken by pivoting from the sophosPID of the process, clicking the `(…)` button next to the sophosPID, and selecting the “Process activity history” query.

Historic web shell file creation events

Alternatively, to identify web shells that have been dropped but may have been deleted, you can interrogate the Sophos process and file journals to look at historic file creations for `.aspx` files in the last day by using the below XDR query for live Windows devices. To increase your hunt time range you can change ‘now’ and ‘-1 days’ to values that needs to be investigated.

```
SELECT
  CAST(datetime(sfj.time, 'unixepoch') AS TEXT) date,
  spj.processName,
  CASE sfj.eventType
    WHEN 0 THEN 'Created'
  END eventType,
  replace(sfj.pathname, rtrim(sfj.pathname, replace(sfj.pathname, '\', '')), '')
fileName,
  spj.pathname processPath,
  sfj.pathname filePath,
  sfj.sophosPID
FROM sophos_file_journal sfj
LEFT JOIN sophos_process_journal spj
  ON spj.sophosPID = sfj.sophosPID
  AND spj.time = replace(sfj.sophosPID, rtrim(sfj.sophosPID, replace(sfj.sophosPID  ,
':', '')), '')/10000000-11644473600
WHERE sfj.time > strftime('%s', 'now', '-1 days')
  AND sfj.eventType IN (0)
  AND sfj.pathname LIKE '%.aspx';
```

Similarly, the sophosPID of suspect processes, especially w3wp.exe, should be pivoted from and the process activity history reviewed to determine other actions the adversary may have taken.

Modified applicationHost.config physicalPaths

Threat actors have also been observed modifying the Exchange configuration, typically located at `C:\Windows\System32\inetsrv\Config\applicationHost.config`, to add new virtual directory paths to obfuscate the location of web shells. These paths are defined in the config under `physicalPath` parameter of a `virtualDirectory` definition. Any entries for web shells should be deleted and the IIS service restarted to reload the config.

The below XDR query for live Windows devices will list all `physicalPath` entries of the `applicationHost.config` file.

```
SELECT grep.*
FROM file
CROSS JOIN grep ON (grep.path = file.path)
WHERE
file.path LIKE 'C:\Windows\System32\inetsrv\Config\applicationHost.config'
AND grep.pattern = 'physicalPath'
```

New and suspicious files in System32

Actors have commonly been dropping malicious executables, via a web shell, to the System32 directory. Recently created .exe files and other suspicious files at this path should be investigated.
E.g. `C:\Windows\System32\createhidetask.exe`
E.g. `C:\Windows\System32\ApplicationUpdate.exe`

The below XDR query for live Windows devices will list all the files currently in the System32 directory.

```
SELECT * FROM FILE WHERE PATH LIKE 'C:\Windows\System32\%'
```

3. Investigate historical command executions

PowerShell and cmd child processes of w3wp

As detailed in the previous section, the presence and use of web shells will result in command executions and other suspicious activity stemming from an IIS Worker Process `w3wp.exe`.

The below query for the XDR Data Lake will list details of hosts where `powershell.exe` or `cmd.exe` are child processes of `w3wp.exe` as well as detail the commands that have been executed.

```
SELECT
 ingestion_timestamp,
 unix_time,
 epoch,
 meta_hostname,
 meta_ip_address,
 meta_mac_address,
 meta_os_name,
 meta_os_platform,
 meta_public_ip,
 cmdline,
 pid,
 name,
 path,
 sophos_pid,
 parent,
 parent_name,
 parent_path,
 parent_sophos_pid,
 username
FROM
  xdr_data
WHERE
  query_name = 'running_processes_windows_sophos'
  AND parent_name = 'w3wp.exe'
  AND (name = 'powershell.exe'
  OR name = 'cmd.exe')
```

Sophos MTR has observed threat actors executing the following commands during ProxyShell incidents which may aid you in identifying post-exploit activity.

- whoami
- Invoke-WebRequest
- Start-Process

- ping
- mkdir
- reg add
- net user
- net accounts
- net localgroup
- icacls
- takeown
- tasklist
- schtasks

4. Locate other forms of persistence

Scheduled Tasks

Sophos has observed threat actors establishing persistence on compromised devices by creating scheduled tasks to periodically execute a suspicious binary. The below XDR query for live Windows devices can be used to list the current Scheduled Tasks on a device which should be reviewed, and any suspicious tasks investigated.

```
SELECT * FROM SCHEDULED_TASKS
```

How Sophos Managed Threat Response (MTR) can help

Threats such as ProxyShell are a great example of the peace of mind you get knowing your organization is backed by an elite team of threat hunters and incident response experts.

When the ProxyShell news broke, the Sophos MTR team immediately began to hunt and investigate in customer environments to determine if any activity was related to the attack. Additionally, they looked to uncover any new artifacts (e.g. IOCs) related to the attack that could provide further protection for all Sophos customers.

The 24/7 nature of Sophos MTR meant that not a single second was wasted as we started hunting for evidence of abuse, ensuring our customers were protected.

Concerned about ProxyShell? Contact Sophos MTR today to ensure that any potential adversarial activity in your environment is identified and neutralized, before any damage is done.

Change log

2021-08-24 UTC 08.00 Added Sophos detections
2021-08-24 UTC 08.41 Fixed error in Exchange version script
2021-08-24 UTC 13.05 Added details for hunting web shells in modified Exchange config
2021-08-24 UTC 13.54 Added link to Naked Security article on Web Shells

2021-08-24 UTC 15.36 Added details of new IPS signature

2021-08-25 UTC 07:55 Added information on additional behavioral-based protection for LockFile

2021-08-27 UTC 14.53 Aligned recommendations with guidance in our Sophos Community post

2021-08-31 UTC 17.12 Added data lake query for historic command executions semming from w3wp.exe

2021-08-31 UTC 21.29 Restructured Sophos XDR guidance and added queries for searching IIS logs for autodiscover.json abuse, and Windows Events for New-MailboxExportRequest abuse

2021-09-07 UTC 14.54 Added additional file path to Web Shells On Disk query

*2021-09-23 UTC 11.26 Updated “*Analyze IIS logs…” query to search over both Aug and Sept