SolarWinds Attack: Sunburst's DLL Technical Analysis

By Fareed Fauzi

Published: 2021-01-21 · Archived: 2026-04-05 19:13:03 UTC

SolarWinds Attack: Sunburst's DLL Technical Analysis

Posted by **Fareed Fauzi**

Introduction

In late 2020, a sophisticated SolarWinds attack that hit organizations through the supply chain has recently been disclosed by various sources. This was done via a compromised version of SolarWinds Orion which we called the backdoor with the name “Sunburst”. Once the update (include the malicious DLL) is installed, the malicious DLL will be imported and loaded by the legitimate SolarWinds.BusinessLayerHost.exe executable.

Sunburst is a trojan version of a digitally signed SolarWinds Orion plugin named ***SolarWinds.Orion.Core.BusinessLayer.dll***. The malicious DLL contains a backdoor code used to initiate a function that will do the communication with the victim’s system via HTTP to the attacker’s command and control server [1]. The malicious code initiation will give full access to the victim which may retrieve and execute commands that instruct the backdoor to transfer files, remote execution, profile victim’s system information, and complete control over the affected system.

Name: SolarWinds.Orion.Core.BusinessLayer.dll
MD5: b91ce2fa41029f6955bff20079468448

File type: Dynamic Link Library

The malicious function code that was being patched in the compromised DLL by the attacker resides in ***OrionImprovementBusinessLayer.Initialize*** which all malicious subfunctions were started right here. The function ***Initialize*** was invoked at line 119 by the parent function ***RefreshInternal*** as shown in Figure 1 below.

Figure 1: Invoking of *Initialize* function

In the ***Initialize*** method in Figure 2 below, we can see that the code trying to check if the current process executable is *solarwinds.businesslayerhost* where the hash of the current process being generated by the function *GetHash.*

Figure 2: Check if the current process is ***solarwinds.businesslayerhost***

The code use function *GetHash* to check the hash of the process. We will see this *GetHash* function often after this as the attacker obfuscate those important strings. Deep diving into the code of the ***GetHash*** will give us ideas how things get going. Looking into the subroutine *GetHash*, the function uses Fowler–Noll–Vo hash (***FNV-1a) + XOR*** algorithm which we can refer to in Wikipedia. Figures 3 and 4 below comparing the algorithm being used.

Figure 3: *GetHash* function

Figure 3: Wikipedia's ***FNV-1a*** explained

The next thing that needs to be explained in the *Initialize* function is at lines 116 to 118 in figure 4 below. At these lines, the malware waits about two weeks/12 days before it executes to avoid any suspicious activity detection.

Figure 4: The malware waits for about 2 weeks to execute

After about 2 weeks, the malware starts to execute the next line where the malware creates the named pipe ***583da945-62af-10e8-4902-a8f205c72b2e*** to ensure only one instance of the backdoor is running.

Figure 5: The sample creates named pipe

In figure 5, after creates the named pipe, the sample check for modes of operation as described by FireEye. If the mode return "***Truncate***", the malware will be terminate and exit.

Figure 6: Makes some delay execution

After the truncate mode being checked and pass, the malware then will delay the execution of the next line about 30min to 120min.

Figure 7: Sunburst check for domain-joined

In figure 7, Sunburst also checks if the victim is joined to an Active Directory domain. Those blacklisted AD domains as follows:

Figure 8: Hashes of blacklisted domain

The next lines of codes will be executed if the current victim does not join the blacklisted AD domains. These encoded strings have been brute-forced by FireEye to determine what are the decoded result of these encoded strings. Refer SolarWinds/SunBurst FNV-1a-XOR hash founds analysis spreadsheet shared by FireEye.

1. swdev.local
2. emea.sales
3. pci.local
4. apac.lab
5. swdev.dmz
6. cork.lab
7. saas.swi
8. dmz.local
9. lab.local
10. dev.local
11. lab.rio
12. lab.brno
13. lab.na
14. test
15. solarwinds

The sample then performs another checking functionality to generate the user ID of the current victim as shown in figures 8 and 9.

Figure 8: ***GetOrCreateUserID*** call

Figure 9: *GetOrCreateUserID* code

In figure 9, the user ID of the victim is built based on 3 values:

1. Network interface MAC address that is up and not a loopback device from the *ReadDeviceInfo* function
2. The domain name that contains in variable ***domain4***
3. ***HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MachineGuid*** value

After that, the user ID is encoded with the XOR MD5 of the value at line 424 to 434 shown in figure 9.

Figure 10: Method ***Update*** being invoke

The backdoor then invokes method ***Update*** which main part of the backdoor resides in here.

Figure 11: Snippet code of the *Update* method

In the first part of the code, as shown in Figure 11, the backdoor begins the domain algorithm generation (DGA) things using class *CryptoHelper*.

Figure 12: Content of CryptoHelper

Sunburst victims, who have been installed and infected by one of the malicious SolarWinds Orion software updates, will query for domain names. The part of the malicious code of the software update will construct and resolve a subdomain of avsvmcloud.com.

The code generates those domain names by taking the victim's User ID and computer's domain name and encoded it with a simple substitution cipher. These encoded strings of subdomains are then being concatenated with one of the following domains to create the hostname to resolve:

- ***.appsync-api.eu-west-1[.]avsvmcloud[.]com***
- ***.appsync-api.us-west-2[.]avsvmcloud[.]com***
- ***.appsync-api.us-east-1[.]avsvmcloud[.]com***
- ***.appsync-api.us-east-2[.]avsvmcloud[.]com***

The example of the generated and encoded C2 domain name as follows:

- ***02m6hcopd17p6h450gt3****.appsync-api.us-west-2.avsvmcloud.com*
- ***06o0865eliou4t0btvef0b12eu1****.appsync-api.us-east-1.avsvmcloud.com*
- ***04spiistorug1jq5o6o0****.appsync-api.us-west-2.avsvmcloud.com*
- ***060mpkprgdk087ebcr1jov0te2h****.appsync-api.us-east-1.avsvmcloud.com*

The subdomains highlighted above are the encoded User ID and computer's domain name which can be decoded using tools from Netresec.

After generated the domain, Sunburst continues invoking another important method called ***UpdateNotification.***

Figure 13: *UpdateNotification* invoked.

Figure 14: Check is there any blacklisted process running

In this method, the backdoor check for the blacklisted processes running based on Figure 14. If there is a blacklisted process running, the backdoor will exit the code. If we deep dive into the method, we can see three important sub-method which are ***SearchAssemblies***, ***SearchServices***, and ***SearchConfiguration***.

Figure 15: Identify AV, EDR, tools, and drivers

The backdoor uses several blocklists to identify anti-virus and endpoint detection response processes, drivers, and services. It also checks for any forensic and malware analysis tools in these three functions.

SearchAssemblies = Processes of malware analysis and forensics tools

SearchServices = Anti-Virus and EDR services

***SearchConfiguration* =** Anti-Virus and EDR drivers

Below figures (Figure 16-19) show the hashes of the blacklisted process of malware analysis and forensics tools, AV and EDR services, and drivers:

Figure 16: The processes hashes included in the field ***assemblyTimeStamps***

Figure 17: List of the blacklisted malware analysis and forensics tools hashes.

Figure 16: The services hashes included in the field ***svcList***

Figure 17: List of the blacklisted AV and EDR services hashes.

Figure 18: The drivers hashes included in the field ***configTimeStamps***

The backdoor retrieves all the driver listing via the WMI query *Select * From Win32_SystemDriver* as shown in figure 18. The drivers hashes are included in the field ***configTimeStamps***.

Figure 19: List of the blacklisted AV and EDR drivers hashes.

All the decoded version of the encoded hashes can be checked here. Thanks to the FireEye team!

Figure 20: Bruteforced blacklist hashes spreadsheet

Next, in the while loop, the sample check for the processes, services, and drivers again. If the victims do not have the indicator of the blacklisted processes, services, and drivers, the backdoor continues to execute the following codes.

Figure 21: Check for the process again

Continue investigation of the code at line 222 as we see the backdoor trying to get the ***AdressFamily*** of the victim and decide its decision in the switch case after that shown in figure 22.

Figure 22: Switch case of socket AddressFamily Netbios

The Command and Control beaconing is starting from here. If the ***AddressFamily*** is NetBios the backdoor will either initiate the C2 beaconing or continue the command and control beaconing which we can see at line 248 in Figure 22 where method **Initialize** being invoked.

Figure 23: C2 things in *Initialize* method

Supported commands for the C2 can be view in the *JobEngine* field as shown as follow in figure 24.

Figure 24: ***JobEngine*** contains the supported command of the Command and Control

Once the Sunburst is gained access to the victim machine, depending on the objectives of the actor, any malicious actions and activities can be executed like stealing sensitive data, source codes, etc.

Conclusion

The cyberattack of this campaign is a highly skilled adversary. The threat actors behind this cyber attack campaign got access to numerous organizations around the world including Malaysia's organizations. Every organization in the world that using SolarWind’s Orion IT monitoring and management software must be alerted with this campaign to take precautions for this matter as the attack still ongoing right now.

IOC

The following SHA256 hashes are associated with Sunburst DLL files:

- e0b9eda35f01c1540134aba9195e7e6393286dde3e001fce36fb661cc346b91d
- a58d02465e26bdd3a839fd90e4b317eece431d28cab203bbdde569e11247d9e2
- 32519b85c0b422e4656de6e6c41878e95fd95026267daab4215ee59c107d6c77

- dab758bf98d9b36fa057a66cd0284737abf89857b73ca89280267ee7caf62f3b
- eb6fab5a2964c5817fb239a7a5079cabca0a00464fb3e07155f28b0a57a2c0ed
- c09040d35630d75dfef0f804f320f8b3d16a481071076918e9b236a321c1ea77
- ffdbdd460420972fd2926a7f460c198523480bc6279dd6cca177230db18748e8
- b8a05cc492f70ffa4adcd446b693d5aa2b71dc4fa2bf5022bf60d7b13884f666
- 20e35055113dac104d2bb02d4e7e33413fae0e5a426e0eea0dfd2c1dce692fd9
- 0f5d7e6dfdd62c83eb096ba193b5ae394001bac036745495674156ead6557589
- cc082d21b9e880ceb6c96db1c48a0375aaf06a5f444cb0144b70e01dc69048e6
- ac1b2b89e60707a20e9eb1ca480bc3410ead40643b386d624c5d21b47c02917c
- 019085a76ba7126fff22770d71bd901c325fc68ac55aa743327984e89f4b0134
- ce77d116a074dab7a22a0fd4f2c1ab475f16eec42e1ded3c0b0aa8211fe858d6
- 2b3445e42d64c85a5475bdbc88a50ba8c013febb53ea97119a11604b7595e53d
- 92bd1c3d2a11fc4aba2735d9547bd0261560fb20f36a0e7ca2f2d451f1b62690
- a3efbc07068606ba1c19a7ef21f4de15d15b41ef680832d7bcba485143668f2d
- a25cadd48d70f6ea0c4a241d99c5241269e6faccb4054e62d16784640f8e53bc
- d3c6785e18fba3749fb785bc313cf8346182f532c59172b69adfb31b96a5d0af
- d0d626deb3f9484e649294a8dfa814c5568f846d5aa02d4cdad5d041a29d5600
- c15abaf51e78ca56c0376522d699c978217bf041a3bd3c71d09193efa5717c71

The following domain names are associated with Sunburst cyber-attack campaign:

- avsvmcloud[.]com
- databasegalore[.]com
- deftsecurity[.]com
- digitalcollege[.]org
- freescanonline[.]com
- globalnetworkissues[.]com
- highdatabase[.]com
- incomeupdate[.]com
- kubecloud[.]com
- lcomputers[.]com
- mobilnweb[.]com
- panhardware[.]com
- seobundlekit[.]com
- solartrackingsystem[.]net
- thedoccloud[.]com
- virtualwebdata[.]com
- webcodez[.]com
- websitetheme[.]com
- zupertech[.]com

Reference

1. https://www.fireeye.com/blog/threat-research/2020/12/sunburst-additional-technical-details.html
2. https://www.fireeye.com/blog/threat-research/2020/12/evasive-attacker-leverages-solarwinds-supply-chain-compromises-with-sunburst-backdoor.html
3. https://docs.google.com/spreadsheets/d/1u0_Df5OMsdzZcTkBDiaAtObbIOkMa5xbeXdKk_k0vWs/edit#gid=0
4. https://labs.sentinelone.com/solarwinds-sunburst-backdoor-inside-the-stealthy-apt-campaign/
5. Colin Hardy videos on Sunburst on Youtube
