Using Splunk to Detect Sunburst Backdoor | Splunk

By Ryan Kovar

Published: 2020-12-14 · Archived: 2026-04-05 12:45:01 UTC

Introduction to Sunburst Backdoor

On Sunday afternoon, FireEye released a report on what they are calling the “Sunburst Backdoor.” I highly recommend you read their phenomenal whitepaper for an in-depth analysis, but here are the basics: an advanced adversary trojanized a legitimate dll of the SolarWinds Orion software and fed that into the Solarwinds' customers’ update cycle. Once infected, this trojanized backdoor allows the adversary to move laterally in a victim’s network and steal their critical data.

At this time, FireEye has detected global activity going back at least to the spring of 2020 with many different verticals targeted. Combined with the recent CISA Emergency Directive 21-01, we felt it was essential to provide a quick response with high-level guidance to our customers to help them detect and protect their networks. Splunk’s research team will provide much more bespoke and customized detections as they work the scenario through our labs in the coming days.

What You Need to Know

The FireEye report reveals that this attack was perpetrated by an advanced adversary who carefully selected targets and changed their attacking infrastructure to match geographical location and even named attacking hosts to match their victims to disguise their traffic better. By using a trusted software partner like Solarwinds Orion, they could utilize Solarwinds' position in the network to spread laterally across on-prem and cloud infrastructures to capture and exfiltrate data.

While Sunburst Backdoor is a sophisticated attack vector, it is still just a trojan on a network with lateral movement. Many of your typical network defense techniques and incident response techniques can be utilized immediately. If you happen to know which hosts on your network are running SolarWinds Orion, start your hunting with those hosts as this is where the adversary gains a foothold. The Sunburst Backdoor should only be effective on those hosts. Still, the added threat here is any lateral movement out from the Orion hosts, using common techniques or credentials harvested from Orion.

Detection in Splunk Enterprise Security

An event like Sunburst is a great time to revisit our blog, “How Do I Add COVID (or Any) Threat Intelligence From the Internet to Splunk Enterprise Security?” on adding threat intelligence quickly to Splunk Enterprise Security (ES). You can simply swap out the “COVID” threat lists with “SUNBURST” threat lists. This blog will help you update your Splunk SIEM with the IOCs currently released from FireEye and give you detection results if any hosts become infected in the future.

My colleague Shannon Davis has already whipped together several local threat intel files for you to ingest into ES using the techniques above!

IOCs: DNS, Hashes, and IPs

First, let's review the IOCs that FireEye kindly released in their GitHub repo. You could go through and make many "OR" statements and look through your DNS, but I have created some quick lookup files that you could use, especially as those IOCs start getting more and more verbose. Take the guidance from my previous blog posts Lookup Before You Go-Go...Hunting. I've also started throwing some lookup files into a GitHub repo, which you can explore independently. Please note, this is based on what has been released by FireEye or other vendors, but it should get you started.

For example, create lookup tables as I indicated in the blogs above or from my Github repository. Then, run a search like below, and you can find hosts that have communicated to the domains so far detected.

```
 index=main sourcetype=stream:*
| lookup sunburstDOMAIN_lookup Domain AS query
| search isBad=TRUE
| stats VALUES(query) AS "Sunburst" by src_ip
```

Just change the search query and lookup file to match what type of IOC you are looking for (domains, IPs, or hashes).

If you detect traffic to these IPs or domains, take a good look at the Snort alerts released by FireEye. If you are collecting any firewall or proxy traffic logs, you might be able to have a better idea of your compromised hosts looking for traffic with these strings in the URL:

```
 /swip/upd/SolarWinds.CortexPlugin.Components.xml
swip/Upload.ashx
/swip/upd/
```

Lateral Movement

Once the adversary has access to the network via the trojanized dll, they can be detected moving laterally to find and exfiltrate information. Although we have not seen the logs, we can safely assume they are still obeying the laws of network traffic. Using the ever handy-dandy Splunk Security Essentials (SSE) tool, I exported several searches into this PDF. You can use these searches either as is or as inspiration for your own. I highly recommend that you start by looking at any suspicious traffic emanating to or from SolarWinds machines in your network.

Sysmon and Named Pipe

One interesting tidbit released in the FireEye report was also the existence of a named pipe. If you are using Sysmon and Splunk please take a look at Event Codes 17 and 18 and the named pipe “583da945-62af-10e8-4902-a8f205c72b2e”. We’ve provided an example search below:

```
 index=windows sourcetype="XmlWinEventLog:Microsoft-Windows-Sysmon/Operational"
(EventCode=17 OR EventCode=18) PipeName=583da945-62af-10e8-4902-a8f205c72b2e
```

We are also seeing some great work done by the community for Sysmon and Splunk queries but have been unable to test them at this time. Take a quick google and your may find some gold!

Azure Active Directory

Microsoft has also determined that adversaries utilizing the Sunburst Backdoor targeted the Azure AD of victims as part of their lateral movement. This was either done via captured administrative passwords or forged SAML tokens. Luckily, Splunk (via the dapper Ryan Lait) has you covered! If you brought your Azure data into Splunk, you can get some great insight into the activity that the adversary may have taken.

The Azure Active Directory Audit data collected by the Microsoft Azure Add-on for Splunk can help hunt some of the techniques leveraged by this actor. These audit logs capture every interaction between users and resources inside Azure. Here are some example searches to detect:

Monitoring For Changes to App Registrations and Service Principals: New Service Principals:

```
 sourcetype="azure:aad:audit" activityDisplayName="Add service principal"
| stats values(activityDisplayName) AS Action, values(initiatedBy.user.userPrincipalName)
AS UPN, values(targetResources{}.displayName) AS Target,
values(targetResources{}.modifiedProperties{}.displayName) AS "Modified Resources",
values(targetResources{}.modifiedProperties{}.oldValue) AS "Old Values",
values(targetResources{}.modifiedProperties{}.newValue) AS "New Values" by correlationId
| fields - correlationId
```

Credentials and certificates added to Apps or Service Principals:

```
 sourcetype="azure:aad:audit" activityDisplayName="Add service principal credentials"
```

Permissions and role assignments added to Apps or Service Principals:

```
 sourcetype="azure:aad:audit" activityDisplayName="Add app role assignment to service principal" OR
activityDisplayName="Add delegated permission grant" OR activityDisplayName="Add application"
| stats  values(initiatedBy.user.userPrincipalName) AS UPN, values(targetResources{}.displayName)
AS Target, values(targetResources{}.modifiedProperties{}.displayName) AS "Modified Resources",
values(targetResources{}.modifiedProperties{}.oldValue) AS "Old Values",
values(targetResources{}.modifiedProperties{}.newValue)
AS "New Values" by correlationId activityDisplayName
| fields - correlationId
```

Use this search to investigate users adding sensitive permissions to app registrations. #ReadMailInAllMailboxes

Apps modified to allow multi-tenant access:

```
 sourcetype="azure:aad:audit" activityDisplayName="Update application" operationType=Update
result=success targetResources{}.modifiedProperties{}.displayName=AvailableToOtherTenants
| table activityDateTime initiatedBy.user.userPrincipalName,
targetResources{}.displayName additionalDetails{}.value
```

Changes to Azure AD Custom Domains:

```
 sourcetype="azure:aad:audit"  activityDisplayName="Add unverified domain" OR
activityDisplayName=*domain* | stats values(activityDisplayName) AS
Action, values(initiatedBy.user.userPrincipalName) AS UPN, values(targetResources{}.displayName)
AS Target, values(targetResources{}.modifiedProperties{}.displayName)
AS "Modified Resources", values(targetResources{}.modifiedProperties{}.oldValue)
AS "Old Values", values(targetResources{}.modifiedProperties{}.newValue)
AS "New Values" by correlationId
| fields - correlationId
```

VPS Hosts

At this time, it is believed that adversaries have utilized geographically relevant (meaning IP addresses will be local to the country of the victim) Virtual Private Servers (VPS) hosts to access victim networks. Although there is no definitive list of these IPs, we recommend that customers review external to internal network traffic to determine if unknown IP addresses have accessed their systems (especially SolarWinds devices) since spring 2020.

TSTAT Searches (Updated!)

As Splunkers around the world have been working to find/detect the Sunburst activity, many of our customers have found that our quick searches above were not scalable and turned to TSTATS to help them cope with the volume in their data models. My colleague Don Slife went ahead and whipped up some queries that you might find useful AND scalable. Please note, you must have CIM compliant data in data models to run these searches.

To find malicious domains in network resolution datamodel This search will look across DNS data in the Network Resolution data model using the sunburstDOMAIN_loopup file above. If you would prefer, remove the subsearch and just look for the avsvmcloud[.]com domain in order to detect the primary IOC.

```
| tstats summariesonly=true earliest(_time) as earliest latest(_time) as latest count as total_conn values(DNS
    [| inputlookup sunburstDOMAIN_lookup
    | rename Domain as DNS.query
    | table DNS.query] OR DNS.query=*avsvmcloud.com by DNS.src DNS.vendor_product DNS.record_type DNS.message_t
| sort earliest
| eval earliest=strftime(earliest, "%c"), latest=strftime(latest, "%c")
```

To find malicious IP addresses in network traffic datamodel This search will look across the network traffic datamodel using the sunburstIP_lookup files we referenced above.

```
| tstats summariesonly=true earliest(_time) as earliest latest(_time) as latest count as total_conn values(All_
    [| inputlookup sunburstIP_lookup
    | rename IP as All_Traffic.dest
    | table All_Traffic.dest] by All_Traffic.src All_Traffic.vendor_product
| sort earliest
| eval earliest=strftime(earliest, "%c"), latest=strftime(latest, "%c")
```

MITRE ATT&CK

The folks at FireEye were kind enough to map their findings to MITRE ATT&CK. Like the lateral movement work above, I went through Splunk Security Essentials and pulled any content we might have associated with the observed tactics and techniques. The PDF ended up being pretty big, but we hope it's useful. If you would rather just pivot in your own SSE instance, here are the 1-1 searches that you should review (I did add anything I thought useful in the PDF, so you might take a peek at it if you have a chance):

- Malicious Powershell Process - Encoded Command
- Anomalous Audit Trail Activity Detected
- Tor Traffic
- Concentration of Attacker Tools by Filename
- Concentration of Attacker Tools by SHA1 Hash
- Sc Exe Manipulating Windows Services
- Prohibited Service Detected
- Prohibited Process Detected
- Anomalous New Service
- Abnormally High Number of Endpoint Changes By User
- First Time Seen Running Windows Service
- Processes with Lookalike (typo) Filenames
- SMB Traffic Allowed
- Anomalous New Process
- High Process Count
- Prohibited Service Detected

Conclusion

We have tried to keep this blog short, sweet and concise. The information above is pulled from our existing products like SSE, ESCU, previous research, and some off the cuff SPL'ing by great Splunkers like Shannon Davis and Ryan Lait. Much (if not all) of the analysis and IOCs above were derived from FireEye and Microsoft blogs on the subject. However, as mentioned above, our threat research team will be releasing more up-to-date information and additional detections as details (and data) become more available.
