
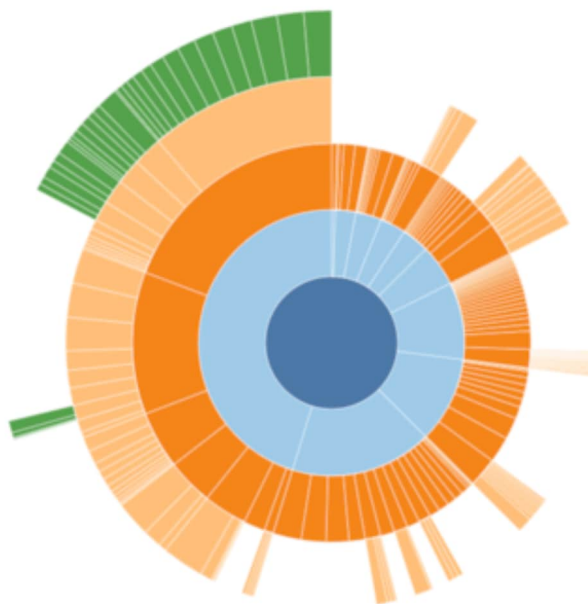


Onboarding Threat Indicators into Splunk Enterprise Security: SolarWinds Continued

 splunk.com/en_us/blog/security/smoothing-the-bumps-of-onboarding-threat-indicators-into-splunk-enterprise-security.html

December 17, 2020



SECURITY



By [John Stoner](#) December 17, 2020

As your team responds to the SolarWinds security advisory, we wanted to provide additional guidance to help you more effectively ingest threat indicators to combat the Sunburst Backdoor malware in [Splunk Enterprise Security \(ES\)](#). After applying these tips, you can jump into this blog, "[Using Splunk to Detect Sunburst Backdoor](#)," and take action.

While the existing method of downloading indicators into Splunk Enterprise has been in place for a long time whether you're running ES 6.4 or older versions like ES 4.5, we have a few tips and tricks to share that will hopefully smooth out some bumps you may have

encountered in the past.

The Issues with Fields

One of the greatest pain points encountered when ingesting threat indicators is the naming of fields. The threat intelligence framework expects that specific header field values are being utilized. The reference for this can be found [here](#).

For example, if I am pulling a list of IP addresses in, the framework is expecting the field that contains the IP address to be called “IP” – not dest, src, source_ip or anything else, just “IP”. Similarly, for a domain name, it is looking for the field domain. Below is an example from the link above.

ip_intel	IP addresses	Local IP Intel	description,ip,weight
domains		Local Domain Intel	description,domain,weight

To illustrate this, we are going to use the [GitHub repository for Sunburst Backdoor](#) that Shannon Davis created as an example. I started off by creating two intelligence downloads called **SunburstDomain** and **SunburstIP**. I have configured the type, description and the URL that it is pulling from.

Name ↕	Type ↕	Description ↕	URL ↕
SunburstDomain	github	Test Load	https://raw.githubusercontent.com/davisshannon/Splunk-Sunburst/main/sunburst-domains.csv
SunburstIP	github	Test Load	https://raw.githubusercontent.com/davisshannon/Splunk-Sunburst/main/sunburst-ipv4.csv

When I click on each one and scroll down, I can see parsing options. Notice in *Fields* we are looking in the first column of the csv file for values that will populate the ip field and the description column will be filled with a constant of SunburstIPListing.

Parsing Options

Delimiting regular expression
A delimiter used to split lines in an intelligence download.

Extracting regular expression
A regular expression used to extract fields from individual lines of an intelligence download.

Fields
A transforms.conf-style expression used to rename or combine fields.

Similarly in our domain download, the domain values are in the first position on the list, which is why we are using **\$1** to denote where to look. We are also using a constant again to describe the list. This time, I have ignored the wisdom of using the supported type of threat intel and instead of calling my first field domain, I am calling it “url_domain”.

Parsing Options

Delimiting regular expression
A delimiter used to split lines in an intelligence download.

Extracting regular expression
A regular expression used to extract fields from individual lines of an intelligence download.

Fields
A transforms.conf-style expression used to rename or combine fields.

While the threat intel download will save and the file will download, I find myself to be disappointed because these domains are not populating my lookups. How can I try to troubleshoot this?

The good news is there are a number of places that can be reviewed along the way to determine what is going wrong.

Tips and Tricks

Start by ensuring that the **Intelligence Downloads** are enabled.

From there, check and see if the file has been downloaded by going to Audit – Threat Intelligence Audit in Enterprise Security. The download_status should show that the threat list has been downloaded. If it didn’t, chances are that the URL is not correct.

Threat Intelligence Audit								Edit	Export ▾	...
Details regarding updates to ES Threat Intelligence										
Download Enabled/Disabled		Download Location								
Enabled ▾		All ▾								
Intelligence Downloads										
_time ↕	stanza ↕	disabled ↕	type ↕	uri ↕	weight ↕	exit_status ↕	download_status ↕			
2020-12-16 08:00:51	local_registry_intel	0	threatlist	lookup://local_registry_intel	1	0				
2020-12-16 08:00:51	local_service_intel	0	threatlist	lookup://local_service_intel	1	0				
2020-12-16 08:00:51	local_user_intel	0	threatlist	lookup://local_user_intel	1	0				
2020-12-16 08:00:51	mozilla_public_suffix_list	0	mozilla_psl	https://publicsuffix.org/list/effective_tld_names.dat	1	0	threat list downloaded			
2020-12-16 19:30:01	SunburstDomain	0	github	https://raw.githubusercontent.com/davisshannon/Splunk-Sunburst/main/sunburst-domains.csv	1	0	threat list downloaded			
2020-12-16 19:28:58	SunburstIP	0	github	https://raw.githubusercontent.com/davisshannon/Splunk-Sunburst/main/sunburst-ipv4.csv	1	0	threat list downloaded			

For skeptics like me, knowing where the threat intelligence files have been downloaded to is a nice piece of information to have. From the command line, go to `$SPLUNK_HOME/etc/apps/SA-ThreatIntelligence/local/data/threat_intel/` directory. If the files have been successfully downloaded, both the SunburstDomain and SunburstIP files can be viewed in the file system. The naming convention of the files are based on the name specified in the Intelligence Downloads.

```
[root@OD-FM-NA-i-0e920acb1c5badb83 threat_intel]# pwd
/four/splunk/etc/apps/SA-ThreatIntelligence/local/data/threat_intel
[root@OD-FM-NA-i-0e920acb1c5badb83 threat_intel]# ll
total 188
-rw----- 1 root root 46047 Dec 16 17:20 emerging_threats_compromised_ip_blocklist.csv
-rw----- 1 root root 114859 Dec 16 19:30 SunburstDomain.csv
-rw----- 1 root root 21998 Dec 16 19:28 SunburstIP.csv
```

Now that I know the files are there, how do I know that they are being parsed and loaded to lookups?

The lower half of the Threat Intelligence Audit dashboard has all of the audit logs pertaining to parsing the threat intelligence and loading of it so that is another nice place to look. That said, it is limited to the last 1000 events and if I want to focus on my two downloads, opening it and then adding my own file names might be a good choice.

`eventtype=threatintel_internal_logs (SunburstIP.csv OR SunburstDomain.csv)`

The screenshot shows a Splunk search interface with the following details:

- Search Query:** `eventtype=threatintel_internal_logs ("SunburstIP.csv" OR SunburstDomain.csv)`
- Results:** 10 events (12/16/20 5:30:55.000 PM to 12/16/20 5:45:55.000 PM)
- Event 1 (Error):**
 - Time:** 12/16/20 5:45:36.717 PM
 - Message:** 2020-12-16 17:45:36,717+0000 ERROR pid=22908 tid=MainThread file=threat_intelligence_manager.py:process_files:551 | status="Exception when processing file." filename="SunburstDomain.csv" message="Parser does not extract a field that can be mapped to a threat intelligence collection."
 - Fields:** eventtype = errOr error eventtype = nix-all-logs eventtype = nix_errors error eventtype = threatintel_internal_logs; host = OD-FM-NA-i-0e920acb1c5badb83.amazonaws.com; source = /four/splunk/var/log/splunk/threat_intelligence_manager.log; sourcetype = threatintel:manager; tag = error
- Event 2 (Info):**
 - Time:** 12/16/20 5:38:36.569 PM
 - Message:** 2020-12-16 17:38:36,569+0000 INFO pid=16194 tid=MainThread file=threat_intelligence_manager.py:write_output:506 | status="Wrote metadata to collection.", collection="threat_group_intel", filename="/four/splunk/etc/apps/SA-ThreatIntelligence/local/data/threat_intel/SunburstIP.csv"
 - Fields:** eventtype = nix-all-logs eventtype = threatintel_internal_logs; host = OD-FM-NA-i-0e920acb1c5badb83.amazonaws.com; source = /four/splunk/var/log/splunk/threat_intelligence_manager.log; sourcetype = threatintel:manager

If I look closely at these two events, I want to highlight what the thrill of victory and the agony of defeat looks like. (Hopefully that resonates with a few of you who grew up watching the Wide World of Sports.) The second event in the image below shows that the IP addresses have been written to the collection. IP addresses are loaded!!!

The first event has thrown an error and this is due to the use of `url_domain` instead of `domain` for our field name and this has caused an exception in processing the file. The framework doesn't know what to do with the downloaded data so it throws an error and moves on. This is definitely something to look for in events and if this error is seen, check the parsing options to ensure the field names used are defined by ES.

```

Event
2020-12-16 17:45:36,717+0000 ERROR pid=22908 tid=MainThread file=threat_intelligence_manager.py:process_files:551 | status="Exception when processing file." filename="SunburstDomain.csv" message="Parser does not extract a field that can be mapped to a threat intelligence collection."
eventtype = error error eventtype = nix-all-logs eventtype = nix_errors error eventtype = threatintel_internal_logs
host = OD-FM-NA-i-0e920acb1c5badb83.amazonaws.com source = /four/splunk/var/log/splunk/threat_intelligence_manager.log
sourcetype = threatintel:manager tag = error

2020-12-16 17:38:36,569+0000 INFO pid=16194 tid=MainThread file=threat_intelligence_manager.py:write_output:506 | status="Wrote metadata to collection.", collection="threat_group_intel", filename="/four/splunk/etc/apps/SA-ThreatIntelligence/local/data/threat_intel/SunburstIP.csv"
eventtype = nix-all-logs eventtype = threatintel_internal_logs host = OD-FM-NA-i-0e920acb1c5badb83.amazonaws.com
source = /four/splunk/var/log/splunk/threat_intelligence_manager.log sourcetype = threatintel:manager

```

At this point I am going to set the domain indicators aside and finish up with the IP address. There are a few intermediate steps that happen that I won't bore you with, the "[Enterprise Security Biology](#)" from .conf2017 covers that in gory detail, but what we are striving for is that the data gets loaded to the KVStore into the various threat collections, depending on the type of indicator.

```
| inputlookup ip_intel | search description=SunburstIPListing
```

description	domain	ip	threat_key	time
SunburstIPListing		107.161.23.204	SunburstIP	1608140316.601231
SunburstIPListing		107.191.99.119	SunburstIP	1608140316.601231
SunburstIPListing		13.56.226.124	SunburstIP	1608140316.601231
SunburstIPListing		13.57.184.217	SunburstIP	1608140316.601231
SunburstIPListing		13.59.205.66	SunburstIP	1608140316.601231
SunburstIPListing		13.65.251.83	SunburstIP	1608140316.601231
SunburstIPListing		13.84.134.105	SunburstIP	1608140316.601231

In this case, I can search the `ip_intel` table to see if the indicators have been loaded. Because I want to focus on the `SunburstIPListing`, which if you recall is the description that I added into the parsing options, I can use a `| search` to narrow my results set to just that list.

From there, the threat intelligence framework takes over and a series of savedsearches run on an on-going basis to correlate events with the different threat collections and write the data to the `threat_activity` datamodel. The correlation search will then run on a scheduled

basis and pluck the new items out of the datamodel and create notables out of them.

I hope this has been helpful and will streamline your onboarding of these threat indicators into your Splunk Enterprise Security instance.



Posted by

John Stoner

•

-

I grew up in Virginia, graduated from Penn State and came back to Virginia where I have worked with databases and cyber for over 20 years. A job or two felt like I really was a cast member of The Office or Office Space, but every one of them taught me something new.