carahsoft.

splunk>

Part 6 | Carahsoft + Splunk Workshop Series

# Data Ingestion

This workshop will go over steps when assessing your data, as well as show you methods of Data Ingestion.

**In this section, we will go over the following:**

- Assessing your needs when getting data in
- Common Splunk Data Ingestion Methods
- Website Monitoring in Splunk

# Assessing Your Needs

The first thing we want to do is stack and assess our needs so that we can determine the best way to get data into a Splunk platform instance. Here are a few questions to help you determine that:

## What kind of data do I want to index?

The type of data you want to index affects how you get data in. If you want to get data in from a proprietary application, maybe using the HTTP Event Collector (HEC) would benefit you which I'll briefly cover later. Or, if you want to ingest Windows data, you might want to use an app to help you get that data in. That leads us to our next question which is...

## Is there an app for that?

Splunk and other third-party developers provide apps that facilitate and improve data ingestion. If there is an app for the type of data you want to get in, you can save yourself the trouble of having to configure and tweak inputs on universal forwarders. We will talk about universal forwarders later. There are many different apps, many of which are free, on Splunkbase.

## Where does the data reside?

For a Splunk Cloud Platform instance, data is always remote, meaning you have to use a universal forwarder or HEC to get the data indexed in. For a Splunk Enterprise instance, data can be local or remote. Examples of local data would be data on a hard disk or solid state drive installed in a desktop, laptop, or server host, data on RAM disk, or data on a resource that's been permanently mounted over a high-bandwidth physical connection that the machine can access at boot time.

Examples of remote resources of data would be network drives on Windows hosts, Active Directory schemas, most cloud-based resources, etc.

# Common Splunk Data Ingestion Methods

## Syslog

One of the most common methods for bringing data into Splunk is syslog. Syslog can be forwarded directly to Splunk, but a best practice is to use a syslog server. There are several advantages to using a Splunk Forwarder in conjunction with a syslog server;

- Splunk's forwarder will buffer the data flow, so there's very little risk of lost data (as opposed to forwarding UDP directly to Splunk).
- Splunk's native capabilities when configuring data inputs are limited to one sourcetype per port. Additional sourcetypes *can* be configured per port, but this adds many steps to the process.
- Utilizing a forwarder/syslog server simplifies the administration of your Splunk environment- add-ons can now be deployed to the forwarder directly through your Splunk GUI.

Splunk is vendor agnostic- it doesn't matter what syslog server you prefer. Carahsoft typically recommends syslog-ng, but any will work.

## API

Splunk uses a REST API that can be utilized to bring in additional data. Splunk's documentation can guide you through this process: https://docs.splunk.com/Documentation/Splunk/8.0.0/RESTREF/RESTprolog

If you are looking to poll REST API's, another option is to use the REST modular input available on Splunkbase:

https://splunkbase.splunk.com/app/1546/

## HTTP Event Collector

HEC enables you to send data over HTTP (or HTTPS) directly to Splunk Enterprise or Splunk Cloud from your application. HEC is token-based, so you never need to hard-code your Splunk Enterprise or Splunk Cloud credentials in your app or supporting files.

## File/Directory Monitoring

Very straight forward, this can quickly and easily be done through the Splunk GUI.

## File Uploads

Also very straight forward, can be done via drag and drop through the GUI.

## Splunk Stream

Splunk Stream is the purpose-built wire data collection and analytics solution from Splunk. Passively capture packets, dynamically detect application, parse the protocol, and send metadata back to your Indexer for over 30 protocols. Detection only for over 300 commercial protocols, even if encrypted.

https://splunkbase.splunk.com/app/1809/

## Other Methods

There are many other methods for capturing data and bringing it into Splunk- this is a key area of versatility.

## Website Monitoring

We've already seen how we can use Splunk forwards to bring logs into Splunk (in our case, we brought in Windows logs, but these could easily come from Linux or another OS).

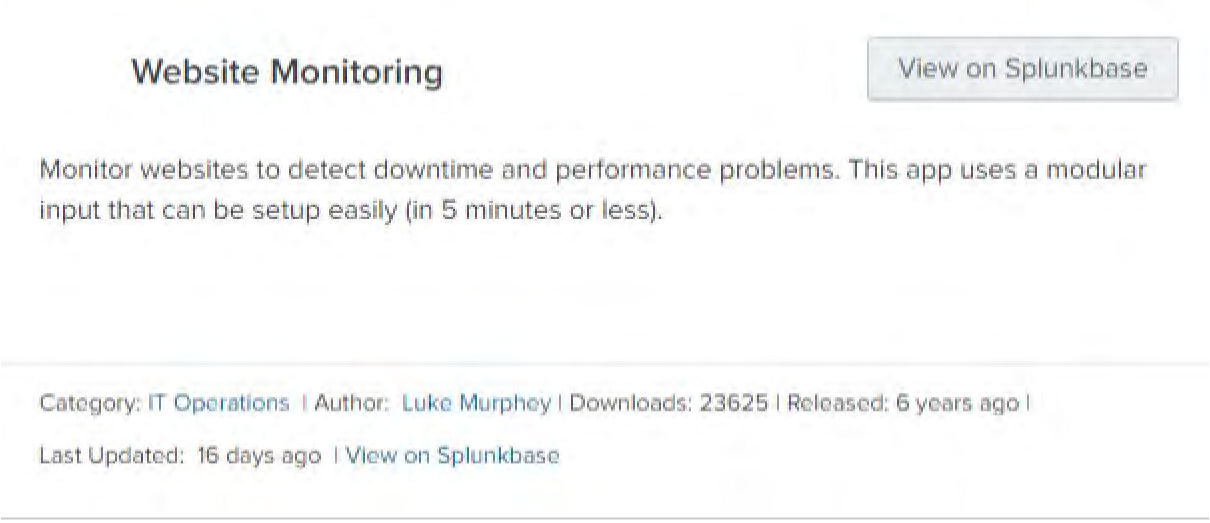
One of the major advantages of Splunk is the ease with which we can look at any data—whether we're forwarding logs from a server, looking at syslog, a .csv, or streaming data.

We'll now look at a few other short examples of what Splunk is able to do. These are only a small sample to give us an idea of Splunk's capabilities- feel free to ask if you have questions about how to ingest data from other sources.

The first example we'll look at is how to use Splunk to monitor a website and ensure it's operating as expected.

In the GUI, go to **Apps>Find More Apps** and enter 'website monitoring input'.

The first option you should see is the 'Website Monitoring' app.



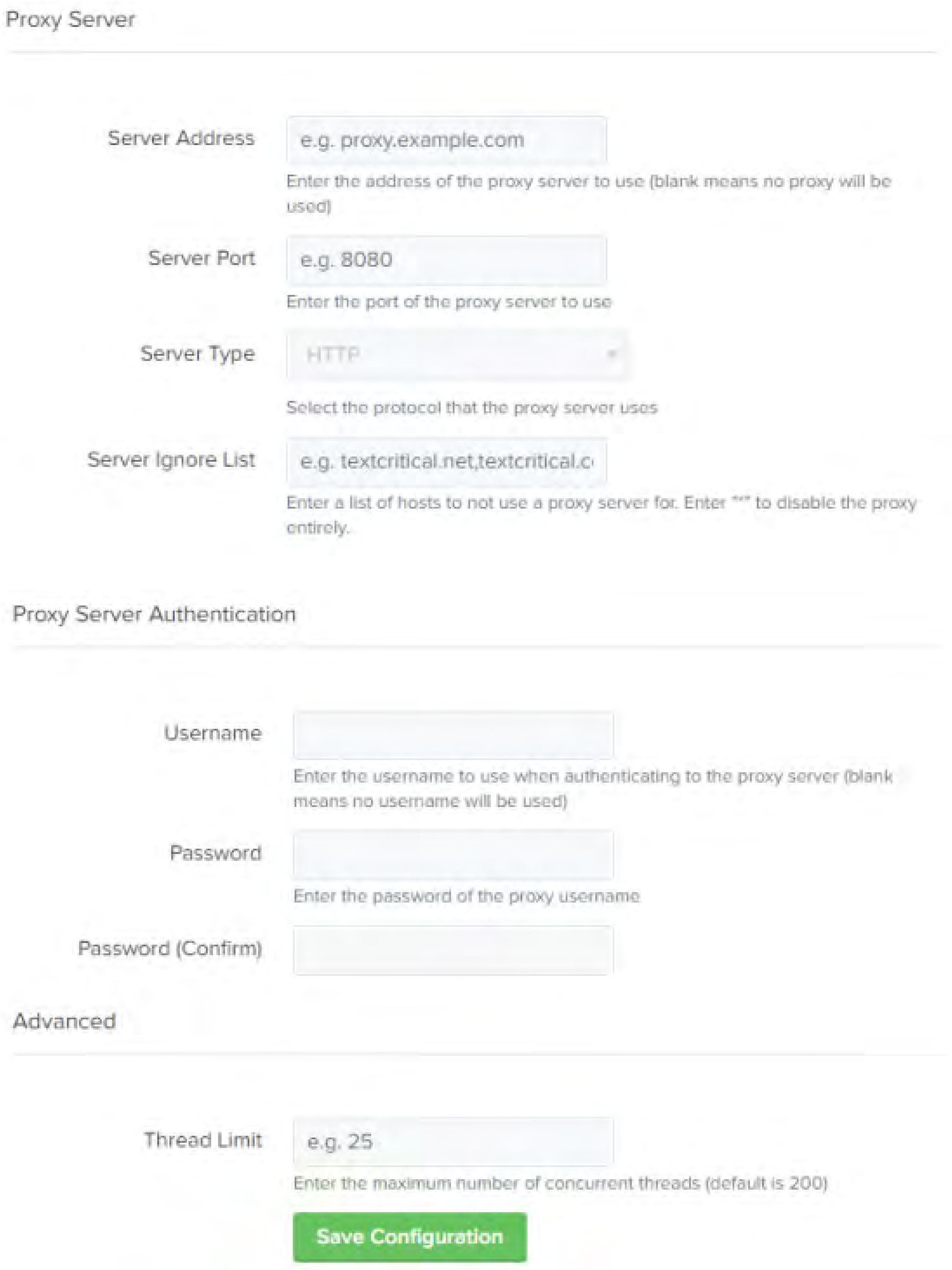
Website Monitoring

Monitor websites to detect downtime and performance problems. This app uses a modular input that can be setup easily (in 5 minutes or less).

Category: IT Operations | Author: Luke Murphey | Downloads: 23625 | Released: 6 years ago |

Last Updated: 16 days ago | View on Splunkbase

Go ahead and download that into your Splunk instance. Once installed, navigate to the appthere are multiple website monitoring apps available- confirm that you see the below before moving on.



Proxy Server

Server Address: e.g. proxy.example.com
Enter the address of the proxy server to use (blank means no proxy will be used)

Server Port: e.g. 8080
Enter the port of the proxy server to use

Server Type: HTTP
Select the protocol that the proxy server uses

Server Ignore List: e.g. textcritical.net,textcritical.c
Enter a list of hosts to not use a proxy server for. Enter "*" to disable the proxy entirely.

Proxy Server Authentication

Username:
Enter the username to use when authenticating to the proxy server (blank means no username will be used)

Password:
Enter the password of the proxy username

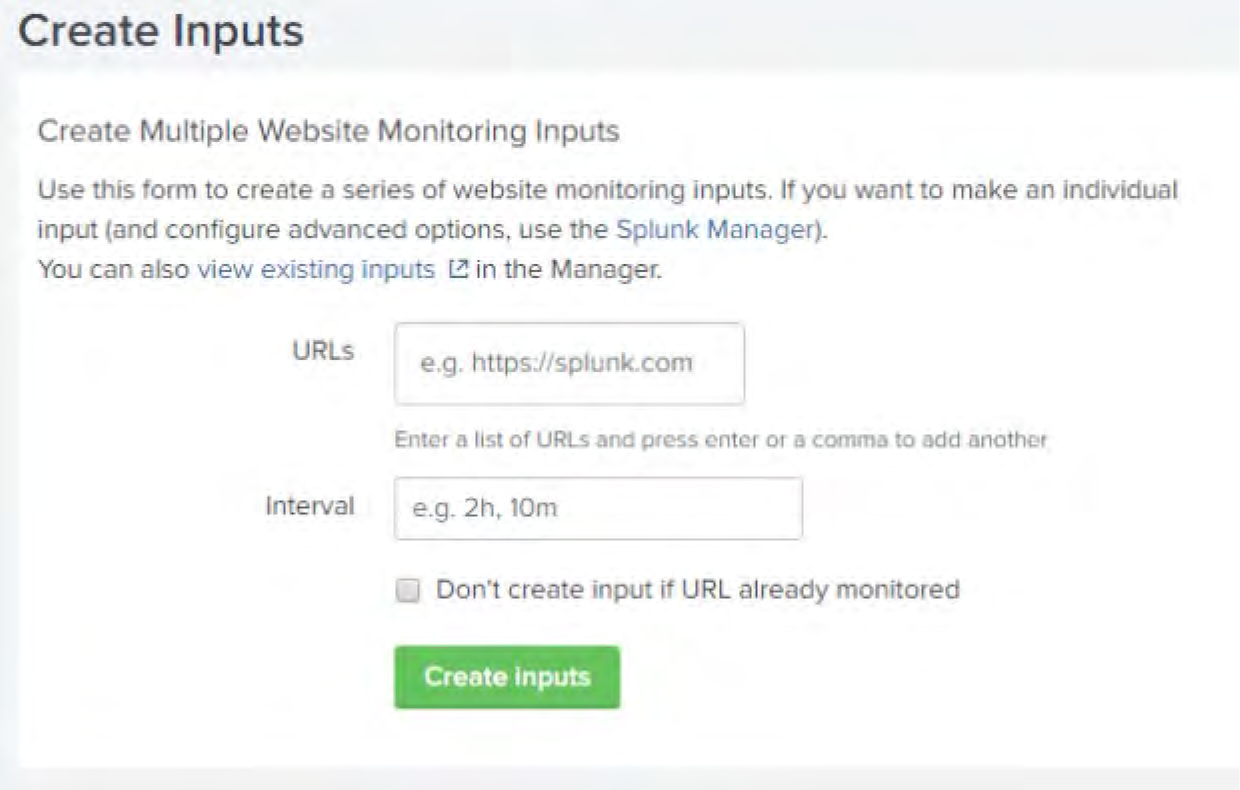
Password (Confirm):

Advanced

Thread Limit: e.g. 25
Enter the maximum number of concurrent threads (default is 200)

The defaults here are fine, and we can **save** this configuration. Now let's create an input so that we can monitor a website.

Navigate to **Create Inputs**; you should see the below. In the URL field, enter **carahsoft.com-** in the interval field enter 2m.



Create Inputs

Create Multiple Website Monitoring Inputs

Use this form to create a series of website monitoring inputs. If you want to make an individual input (and configure advanced options, use the Splunk Manager).
You can also view existing inputs in the Manager.

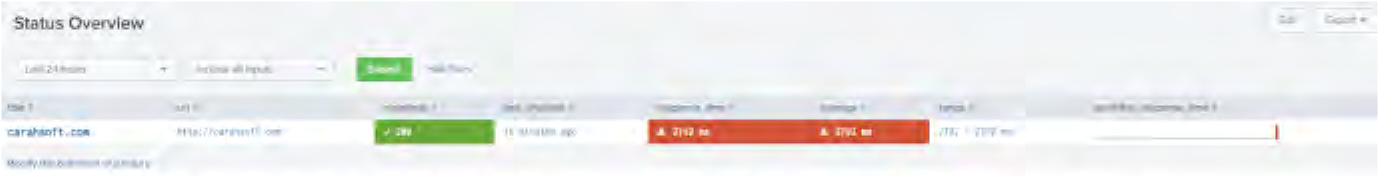
URLs: e.g. https://splunk.com

Enter a list of URLs and press enter or a comma to add another

Interval: e.g. 2h, 10m

□ Don't create input if URL already monitored

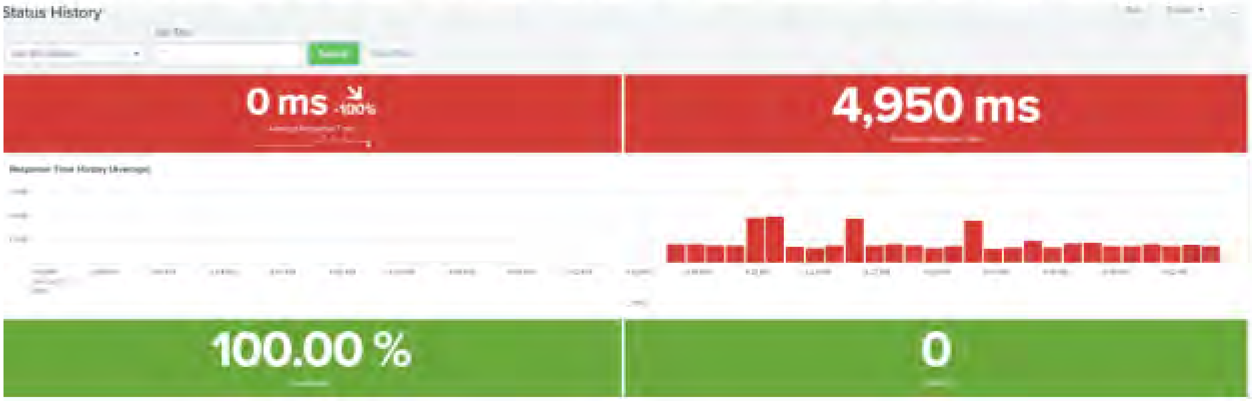
Once you've created this input, navigate to **Status Overview**—you should be getting data in telling you when the website was last checked and givin



Status Overview

| | | | | | | |
|---|---|---|---|---|---|---|
| carahsoft.com | [illegible] | ✓ 200 | [illegible] | ⚠ 2112 ms | ⚠ 2112 ms | [illegible] |

**Status History** will give us a more detailed report of how the website has been performing



Status History

0 ms -100%

4,950 ms

Response Time History (Average)

100.00 %

0

g

information about the response time.