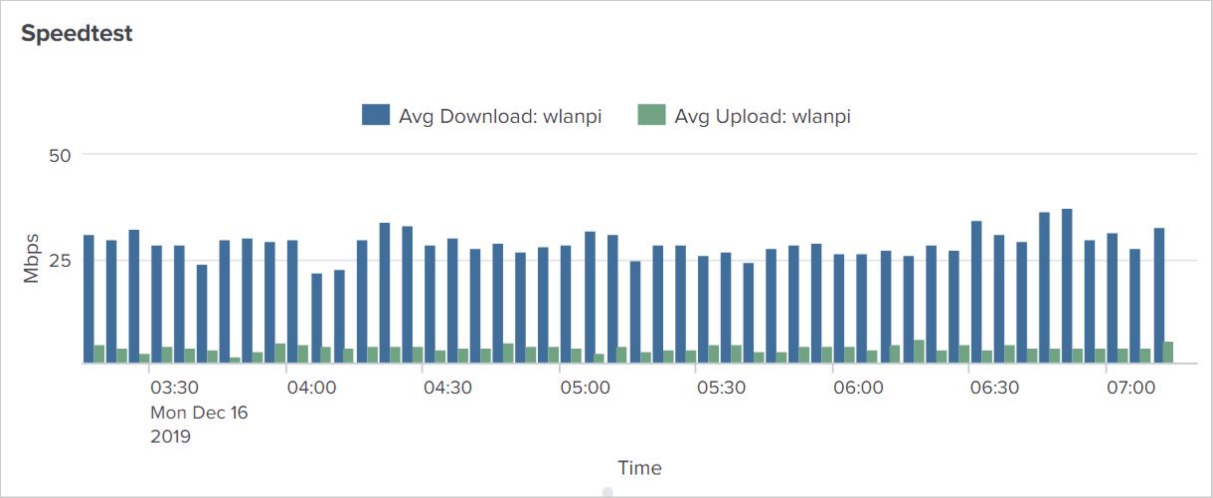
Wiperf - Splunk Build

1 Background

The WLANPi Wiperf mode allows the WLANPi to become a standalone probe that will join a wireless network and run a series of performance tests that provide indications about how well its connection is performing. This data may be used, in conjunction with other data sources, to gain an understanding of the health of the wireless environment.

For more information about Wiperf on the WLANPi, please visit the project GitHub page at : https://github.com/wifinigel/wiperf



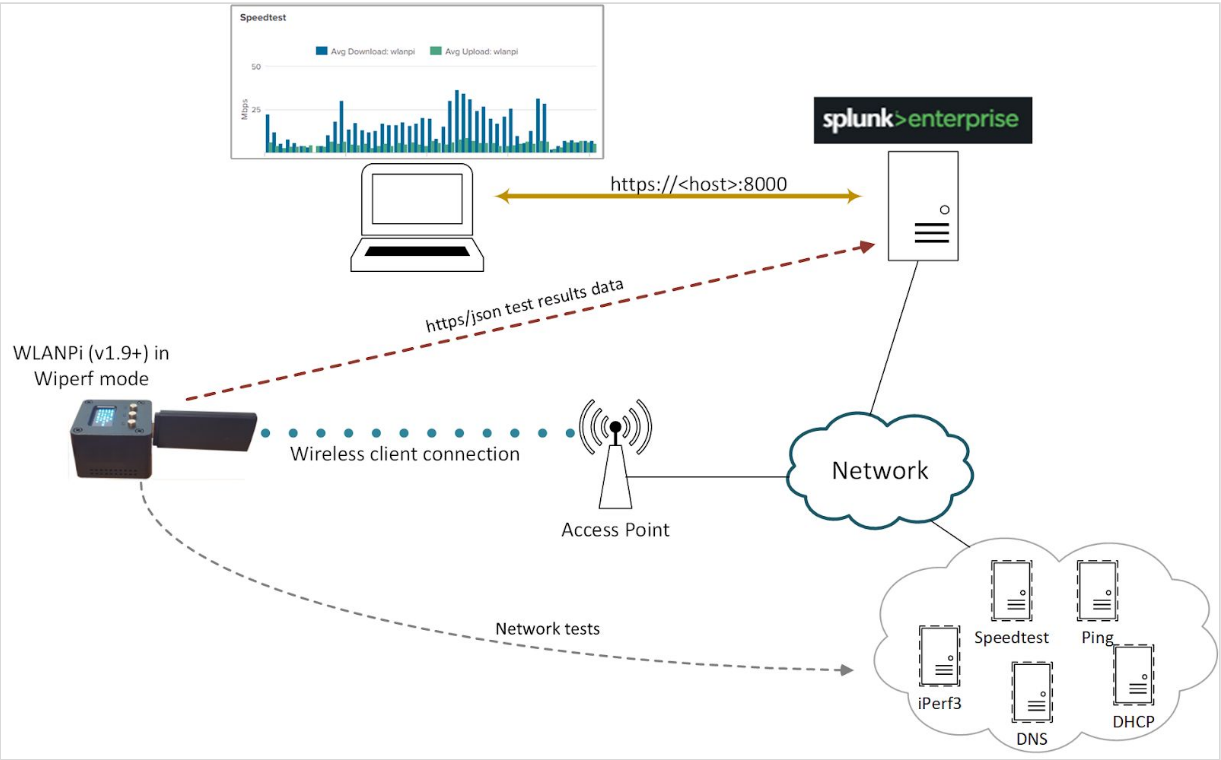
The performance tests that can be run include:

- Speedtest
- Ping (to multiple destinations)
- Iperf3 TCP test
- Iperf3 UDP test
- DNS lookup tests (multiple name queries)
- DHCP renew test

To visualize the performance data gathered by the WLANPi, a reporting server is required to receive the data and visualize it using a reporting package. In this paper we look at how WLANPi data can be sent to a Splunk server and a report created to visualize that data.

Splunk is a powerful data analytics tool that can take data from a variety of sources and allow it to be analyzed and visualized. In the case of the WLANPi, we generate data from the performance tests it runs and then send it over an https connection to a Splunk server in a JSON data format.

The interaction of all components is summarized below:



The operation is summarized as follows:

- The WLANPi is configured as a wireless network client and joins the network under test
- The WLANPi regular runs a series of configurable network tests (controlled by a cron job within the WLANPi)
- After each test, the results data is sent over an https connection from the WLANPi to a Splunk server
- The Splunk server analyzes the data and makes it available via a "dashboard"

- A user who needs to view the performance data from the WLANPi logins in to the reporting GUI of the Splunk server and reviews the performance dashboard

2 Installation

To collect and view the test results data, an instance of Splunk is required. Splunk is a very flexible data collection and reporting package that can take data sent by the WLANPi and present it in a very nice report format. Splunk can be installed on a wide variety of platforms that can be viewed at : https://www.splunk.com/en_us/download/splunk-enterprise.html

This guide does not cover all installation details of the software package, these may be obtained when downloading and installing the software. Note that a free account sign-up is required when downloading the software from the link listed above.

To install Splunk and use it with a handful of probes, a modest server may be built (e.g. I use a low-end Intel NUC), so for testing purposes, don't get too hung up on sourcing a high end server.

The product being installed is Splunk Enterprise. This is a paid-for product, but it has a free-tier for low data volumes (500Mbytes per day). Install initially with all the licensing defaults and then drop back to the free-tier once the eval licence expires a few weeks later. The free tier is plenty for the low volume rates that the WLANPi generates when deploying probes at small-scale.

3 Connectivity Planning

One area to consider is connectivity between the WLANPi and the Splunk instance. The WLANPi needs to be able to access the Splunk server to send its data. If the WLANPi probe is being deployed on a wireless network, how is the performance data generated going to get back to the Splunk server?

If the probe is being deployed on a customer network to perform temporary monitoring, it will obviously join the wireless network under test. But how is the WLANPi going to send its data to the Splunk server ? Many environments may not be comfortable with hooking up the WLANPi to their wired network, hence (potentially) bridging wired and wireless networks. Therefore, in many instances an alternative is required (e.g. send the results data over the wireless network itself out to the Internet to a cloud instance or via a VPN solution such as Zerotier.

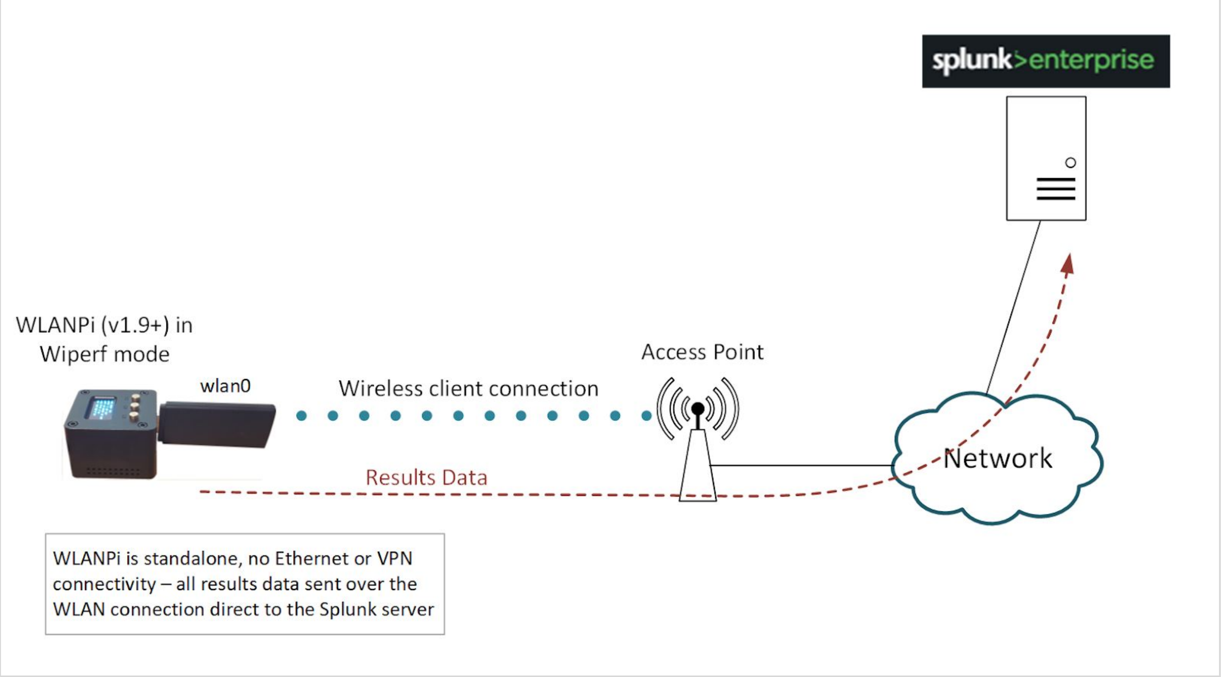
Three topology deployment options are supported:

1. Results data over wireless
2. Results data over Ethernet
3. Results data over VPN/wireless

The method used is configured on the WLANPi probe prior to flipping it in to Wiperf mode by configuring its config.ini file. Is is important to understand the (viable) connectivity path prior to deploying both the probe and the Splunk server.

The 3 connectivity options are discussed below.

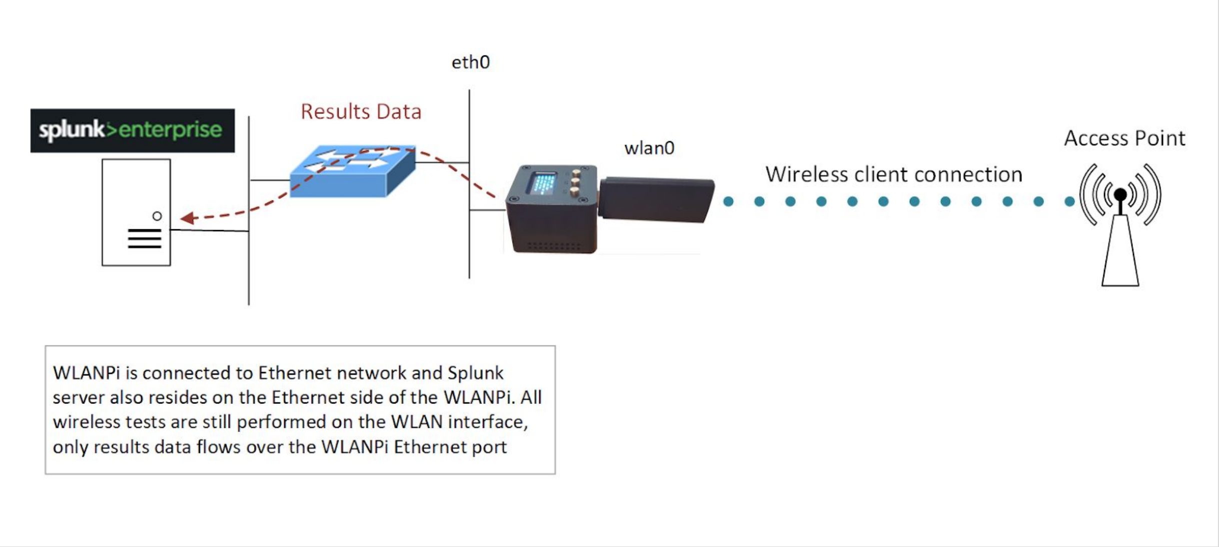
3.1 Results Data Over Wireless


splunk>enterprise
WLANPi (v1.9+) in Wiperf mode
wlan0
Wireless client connection
Access Point
Results Data
Network
WLANPi is standalone, no Ethernet or VPN connectivity – all results data sent over the WLAN connection direct to the Splunk server

In this topology the WLANPi is configured to join an SSID that has the Splunk server accessible via its WLAN interface. Typically, the Splunk server will reside in a cloud or perhaps on a publicly accessible VPS. The WLANPi will continually run the performance tests over the wireless connection and then upload the results directly to the Splunk server over the WLAN connection.

Config.ini settings:

3.2 Results data over Ethernet



WLANPi is connected to Ethernet network and Splunk server also resides on the Ethernet side of the WLANPi. All wireless tests are still performed on the WLAN interface, only results data flows over the WLANPi Ethernet port

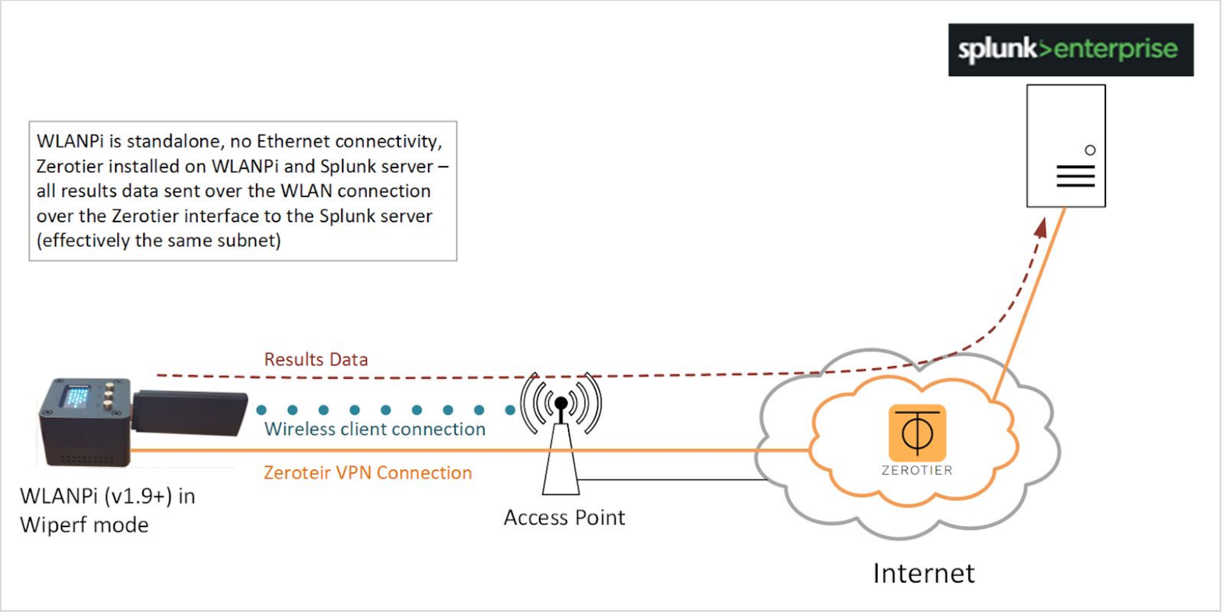
If the Splunk server is being run on the inside of a network environment, it may be preferable to return results data via the Ethernet port of the WLANPi. This topology also has the advantage of results data not being impacted if there are wireless connectivity issues on the WLANPi WLAN connection. To achieve the correct traffic flow, a static route for management traffic is injected into the route table of the WLANPi to force results data over the Ethernet port. Note that the Ethernet port must be connected and up when switching the WLANPi into Wiperf mode for routing to work correctly.

Config.ini settings:

Security notes:

- to prevent traffic leaking between the WLAN and Ethernet interfaces of the WLANPi, IP forwarding is not enabled on the WANPi
- In Wiperf mode, the WLANPi internal firewall allows only connectivity over port 22 incoming over the ethernet and wlan interfaces - all other ports are blocked

3.3 Results data over Zerotier/wireless


splunk>enterprise
WLANPi is standalone, no Ethernet connectivity, Zerotier installed on WLANPi and Splunk server – all results data sent over the WLAN connection over the Zerotier interface to the Splunk server (effectively the same subnet)
Results Data
Wireless client connection
Zeroteir VPN Connection
WLANPi (v1.9+) in Wiperf mode
Access Point
ZEROTIER
Internet

A very simple way of getting the WLANPi talking with your Splunk server is to use the Zerotier service to create a virtual network. In summary, both the Splunk server and WLANPi have the Zerotier client installed. Both are then added to your Zerotier dashboard (by you) and they start talking! Under the hood, both devices have a new virtual network interface created and they connect to the Zerotier cloud-based network service so that they can communicate on the same VLAN in the cloud. As they are on the same subnet from a networking perspective, there are no routing issues to worry about to get results data from the WLANPi to the SPlunk server.

Zerotier has a free subscription tier which allows up to 100 devices to be hooked up without having to pay any fees, It’s very easy to use and get going, plus your Splunk server can be anywhere! (e.g. on your laptop at home). Both devices need access to the Internet for this solution to work.

You can sign up for free, create a virtual network and then just add the IDs that are created by the Splunk server and WLANPi when the client is installed.
Seriously, give it a go...it's quicker to try it than me explaining it here:
https://www.zerotier.com/

Config.ini settings:

To install Zerotier on the WLANPi (or an Ubuntu server), enter the following:

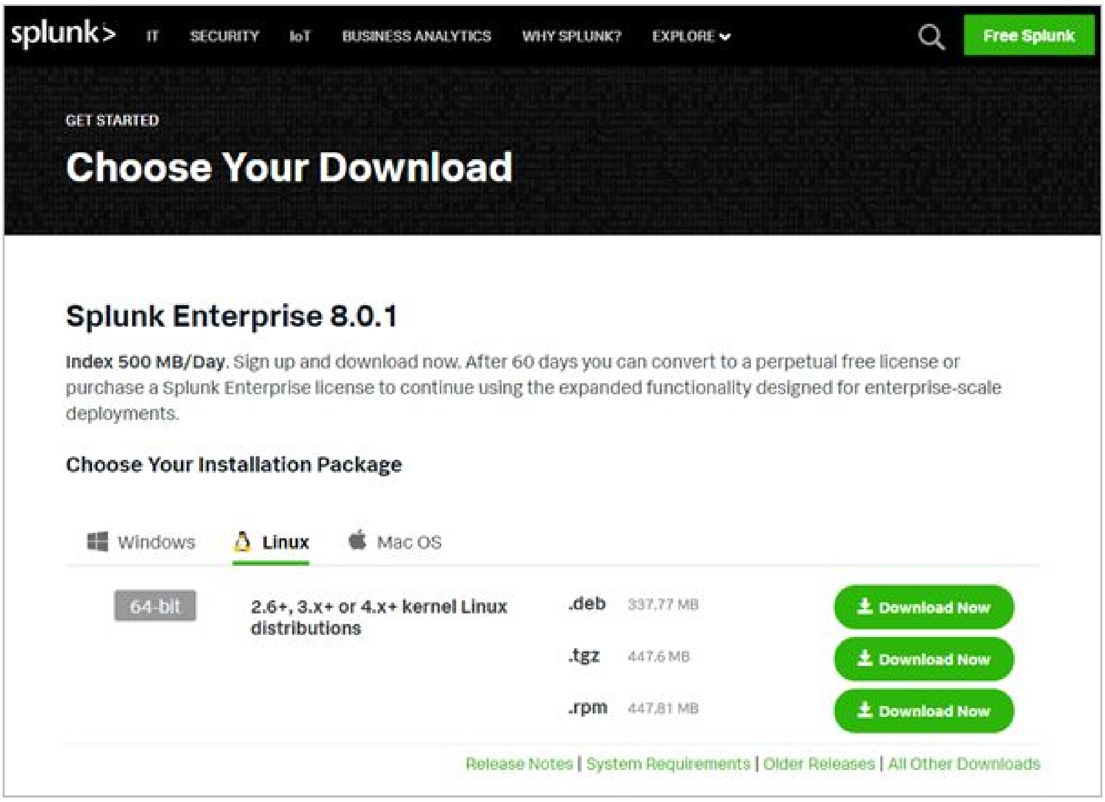
```
                          network number from your Zerotier dashboard
```

4 Basic Splunk Build

Here are the basic steps you will need to install Splunk once you have built your Splunk server hardware/instance.

4.1 Download

Get along to the Splunk web site and sign up for an account if you don't already have one: https://www.splunk.com/en_us/download/splunk-enterprise.html
Once you're logged in to the Splunk site, you'll have a number of OS options, so go ahead and choose your OS option. In the examples that follow, I show Linux commands as that was the server I had available during this build (an Ubuntu machine):



GET STARTED

Choose Your Download

Splunk Enterprise 8.0.1

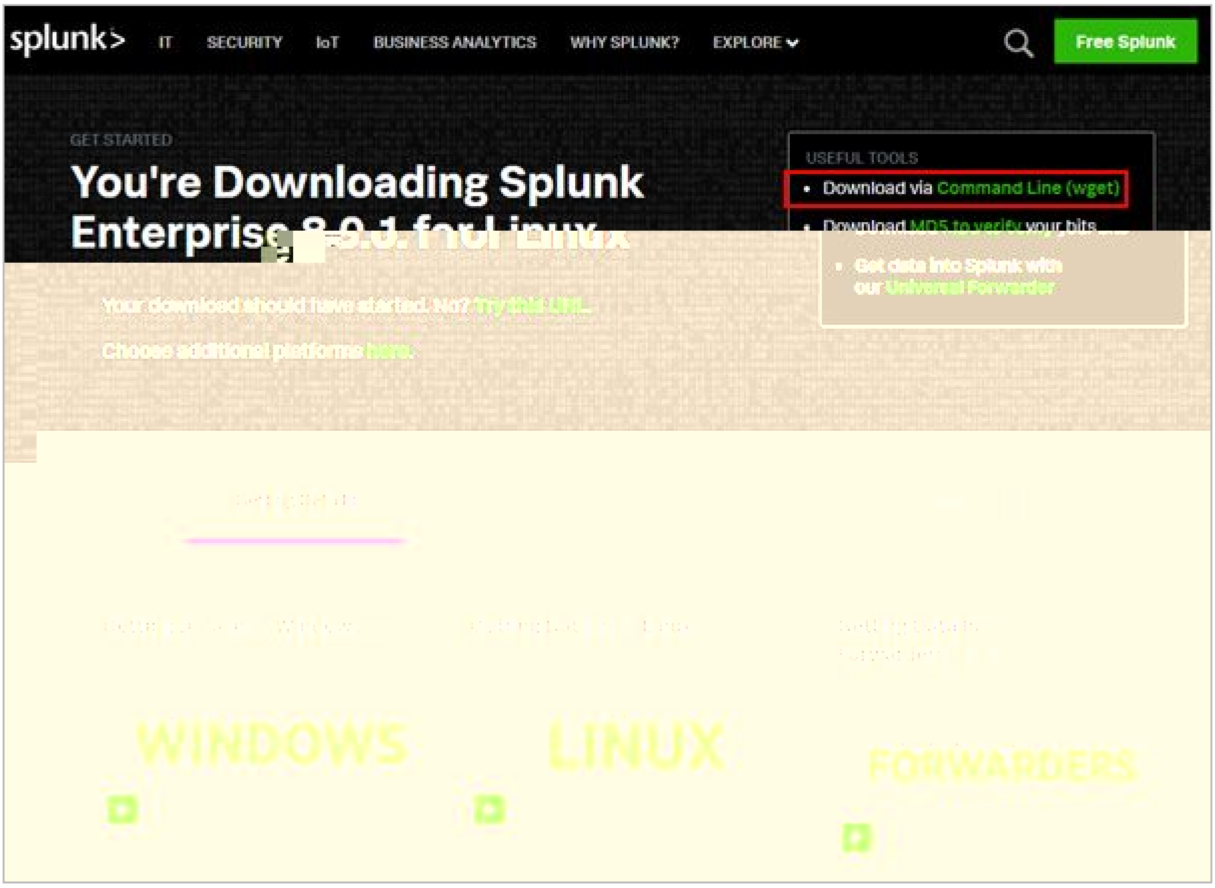
Index 500 MB/Day. Sign up and download now. After 60 days you can convert to a perpetual free license or purchase a Splunk Enterprise license to continue using the expanded functionality designed for enterprise-scale deployments.

Choose Your Installation Package

64-bit	2.6+, 3.x+ or 4.x+ kernel Linux distributions	.deb	337.77 MB	Download Now
		.tgz	447.6 MB	Download Now
		.rpm	447.81 MB	Download Now

Release Notes | System Requirements | Older Releases | All Other Downloads

Once you have hit the download button, Splunk Enterprise will start to download:



GET STARTED

You're Downloading Splunk Enterprise 8.0.1 for Linux

Your download should have started. No? Try this URL.

Choose additional platforms here.

USEFUL TOOLS

- Download via Command Line (wget)
- Download MD5 to verify your bits
- Get data into Splunk with our Universal Forwarder

It’s worth checking the download page to see if there are further download options. If you check the graphic above, you can see there is a “Download via Command Line (wget)” option, which is a much easier way to get the code directly on to your server. The options you will see here will vary between OS selections.

4.2 Installation

Once the software is downloaded, follow the instructions that are appropriate for your OS in the Splunk installation manual (you can look for the latest guides via Google, but at the time of writing it was this manual):

https://docs.splunk.com/Documentation/Splunk/8.0.1/Installation/Chooseyourplatform

Note that while compiling this guide I was performing a build on a Debian Linux platform so made a note of the required steps and have included them in green dialog boxes, like the one shown below, throughout this guide.

For a Debian Linux server use the following:

4.2.1 Starting Splunk

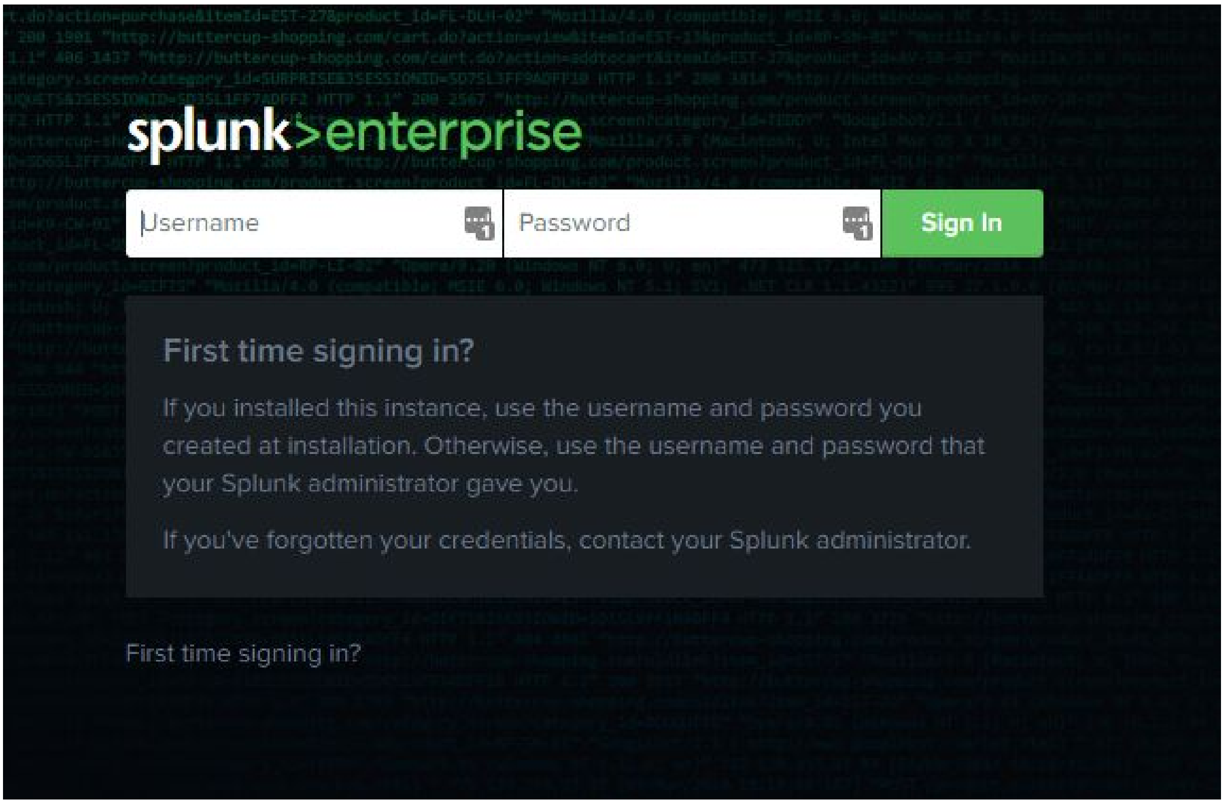
Go to the ‘Next Steps’ section of the installation manual and find out how to start Splunk for the first time and create admin credentials.

On a Debian Linux server, enter the following:

Accept the license agreement and enter a Splunk admin name and password to create an administrator account for your Splunk application.

The web interface for the Splunk server will now be started and the URL of the web GUI shown in the startup dialogs. You may now login to the Splunk web GUI using the admin credential created above. The web GUI is accessed at the URL:

```
http://<Splunk_server_IP>:8000
```



First time signing in?

If you installed this instance, use the username and password you created at installation. Otherwise, use the username and password that your Splunk administrator gave you.

If you've forgotten your credentials, contact your Splunk administrator.

First time signing in?

4.2.2 Automating Startup

Finally, ensure you follow the instructions in the installation manual to enable the Splunk application to auto start if the server is rebooted.

On a Debian Linux server, enter the following:

Basic server installation is now complete.

5 Customizing Splunk

5.1 Configure Data Input To Splunk

We need to tell Splunk how we'll be sending the data from our WLANPi probe in to Splunk. We need to configure a data input that will prepare SPlunk to receive the data and generate an authorization key to be used by the WLANPi when sending data.

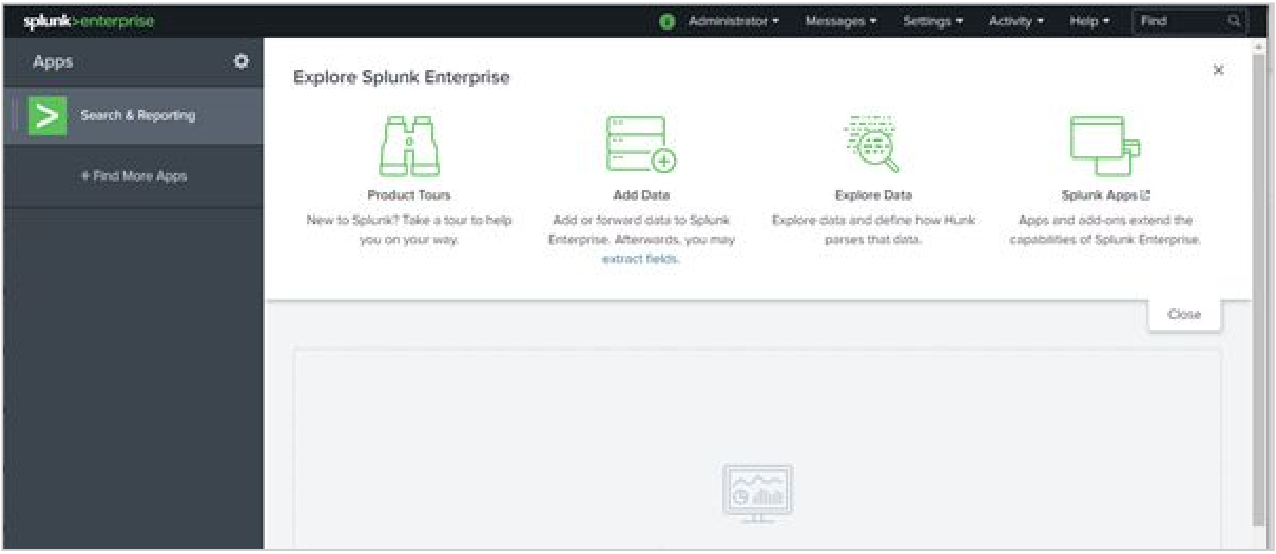
5.1.1 Log In To Splunk

The first step is to login to Splunk using the credentials created during the Splunk install. The URL to use is:

```
http://<Splunk_server_IP>:8000
```

5.1.2 Configure HTTP Event Collector Global Options

After login, the following page will be seen



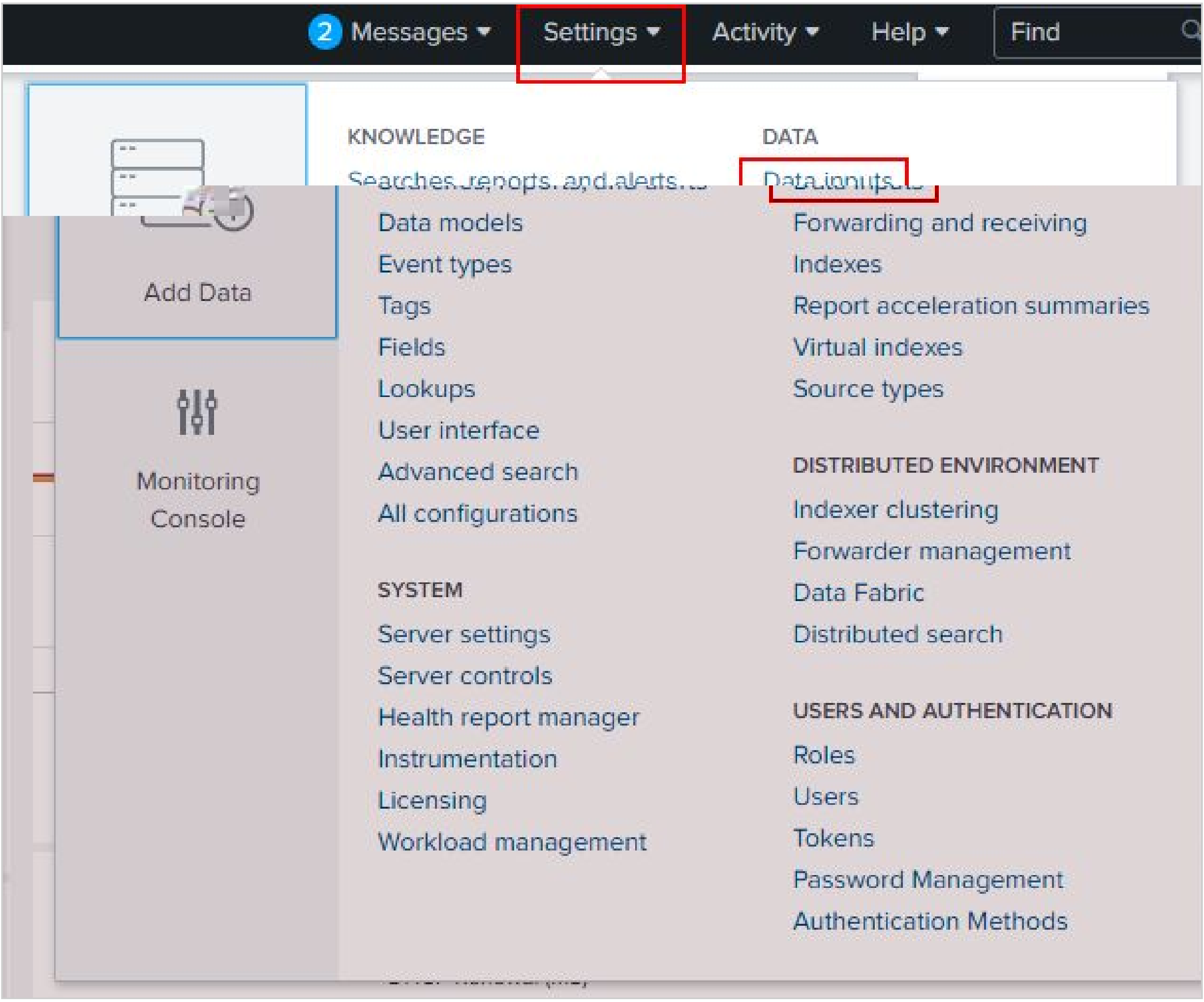
New to Splunk? Take a tour to help you on your way.

Add or forward data to Splunk Enterprise. Afterwards, you may extract fields.

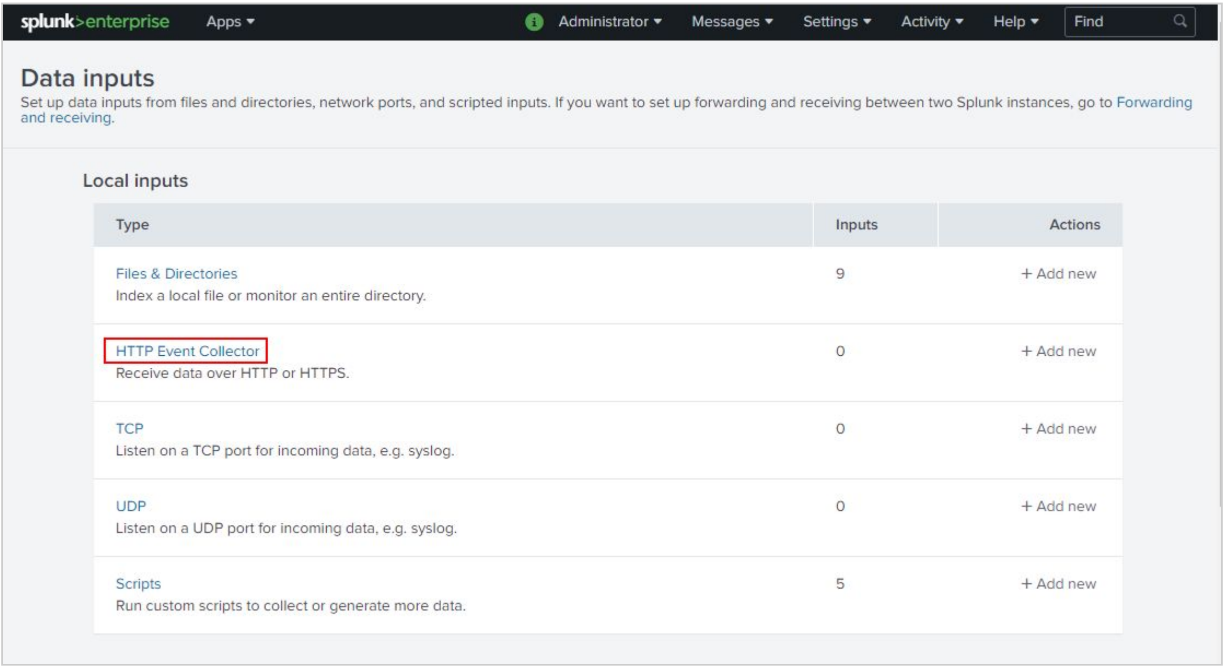
Explore data and define how Hunk parses that data.

Apps and add-ons extend the capabilities of Splunk Enterprise.

Follow the “Settings > Data > Data Inputs” menu options :



Click on the HTTP Event Collector link in the Data Inputs page shown:

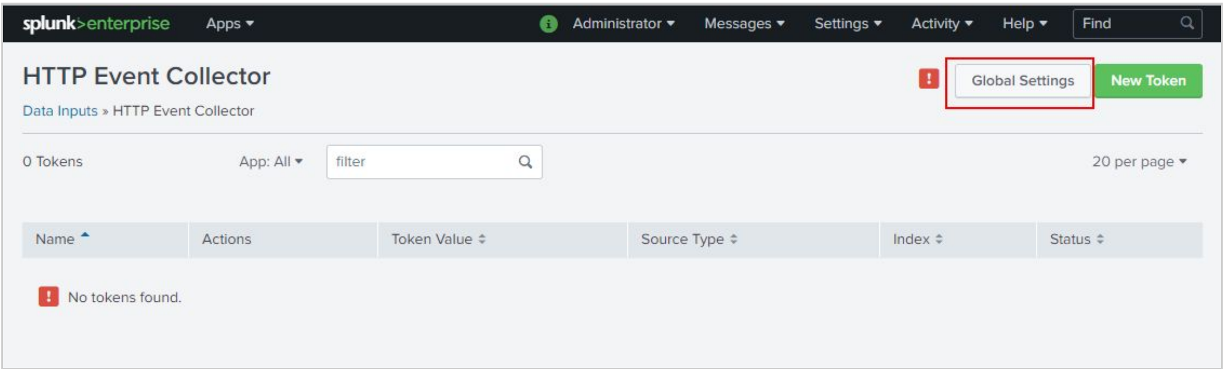


Data inputs

Set up data inputs from files and directories, network ports, and scripted inputs. If you want to set up forwarding and receiving between two Splunk instances, go to Forwarding and receiving.

Local inputs

Type	Inputs	Actions
Files & Directories Index a local file or monitor an entire directory.	9	+ Add new
HTTP Event Collector Receive data over HTTP or HTTPS.	0	+ Add new
TCP Listen on a TCP port for incoming data, e.g. syslog.	0	+ Add new
UDP Listen on a UDP port for incoming data, e.g. syslog.	0	+ Add new
Scripts Run custom scripts to collect or generate more data.	5	+ Add new



HTTP Event Collector

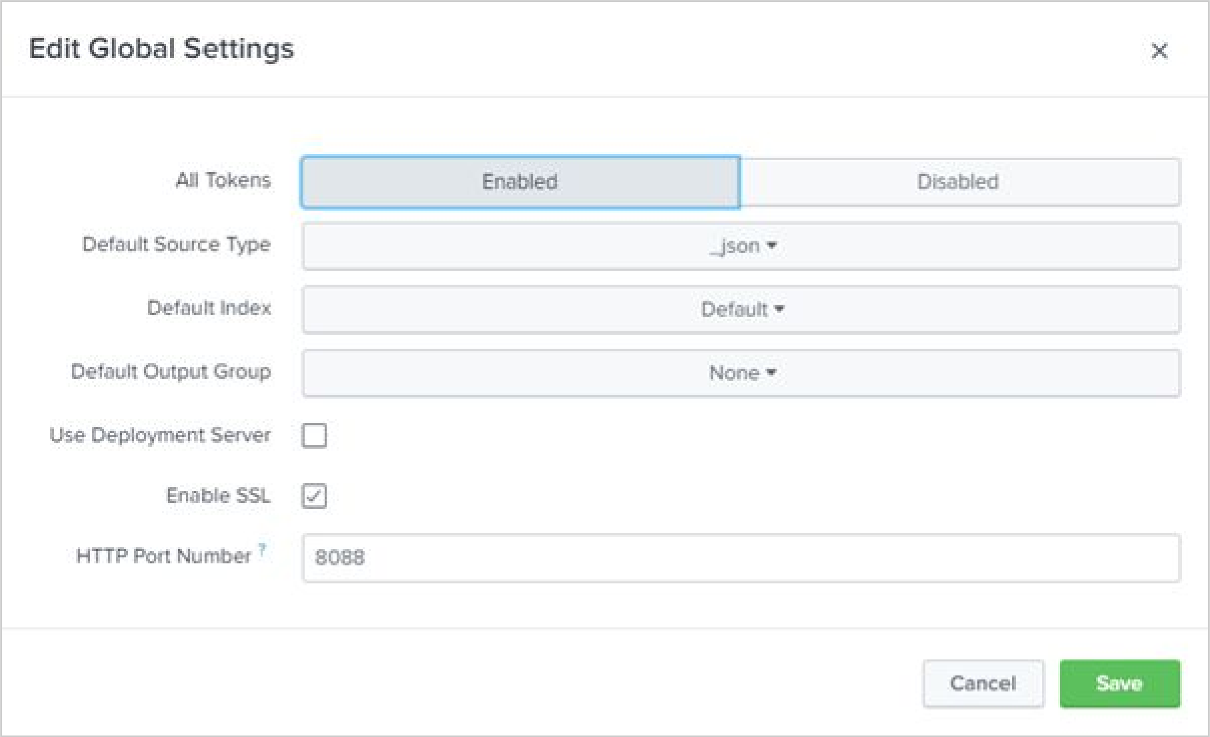
Data Inputs » HTTP Event Collector

0 Tokens

Name	Actions	Token Value	Source Type	Index	Status

No tokens found.

Click on the “Global Settings” button as indicated in the graphic above to reveal the global configuration panel for the HTTP Event Collector:



Edit Global Settings

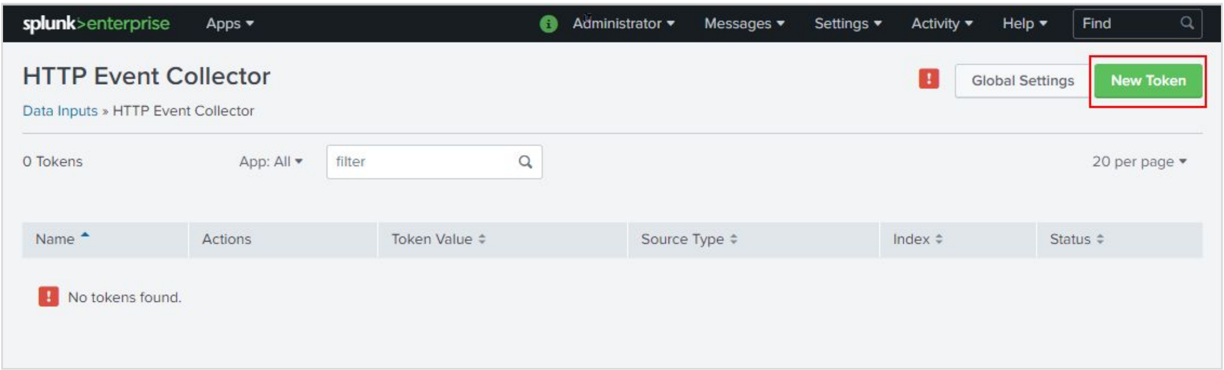
All Tokens	Enabled / Disabled
Default Source Type	_json
Default Index	Default
Default Output Group	None
Use Deployment Server	□
Enable SSL	☑
HTTP Port Number ?	8088

Ensure the panel is configured to look like the example shown above. This should require the following steps:

- **Make sure you hit the All Tokens > Enable button** (disabled by default which stops everything working)
- Default Source Type: Structured > _json
- Hit: Save to take you back to the HTTP Event Collector page

5.1.3 Create a HEC Token

After returning to the HTTP Event Collector page, hit the New Token button. This will start a token creation wizard.



HTTP Event Collector

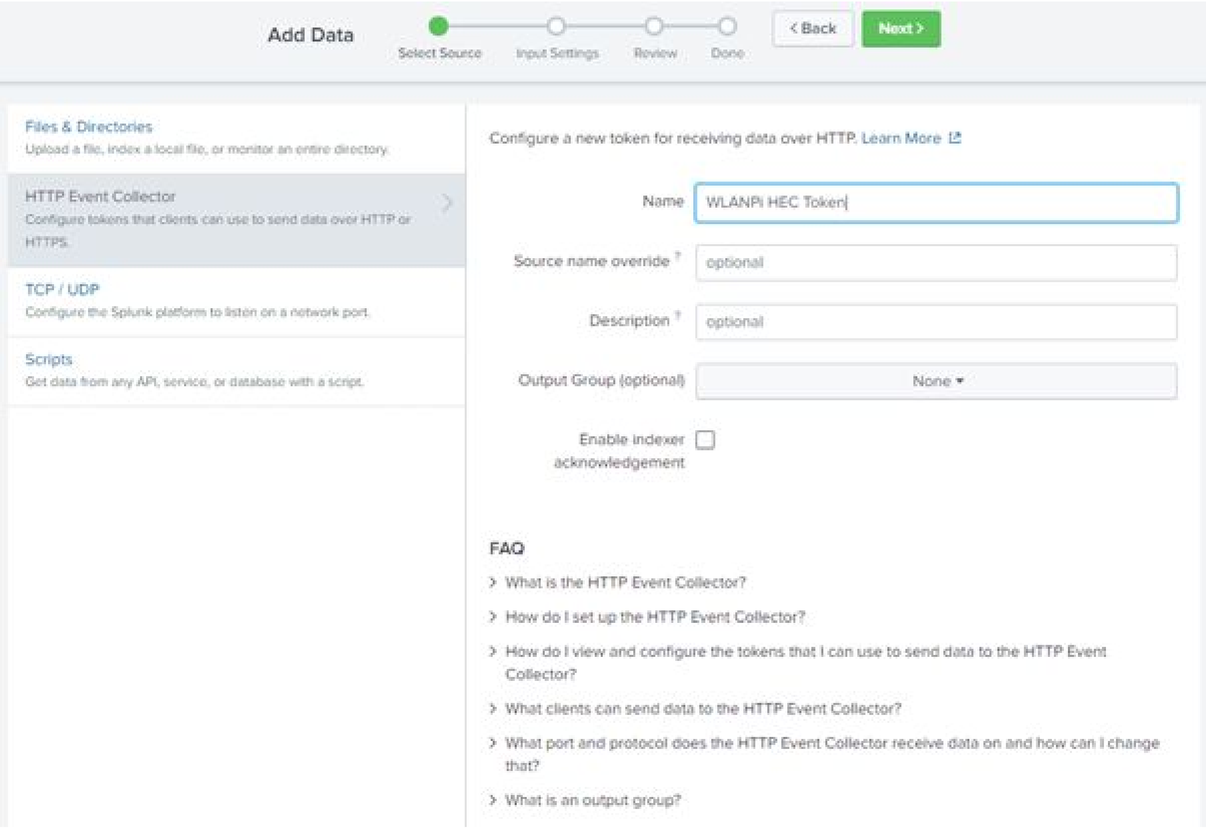
Data Inputs » HTTP Event Collector

0 Tokens

Name ▴	Actions	Token Value ⇕	Source Type ⇕	Index ⇕	Status ⇕

! No tokens found.

Enter a name for the token (WLANPi HEC Token) then hit Next > :



Files & Directories
Upload a file, index a local file, or monitor an entire directory.

HTTP Event Collector
Configure tokens that clients can use to send data over HTTP or HTTPS.

TCP / UDP
Configure the Splunk platform to listen on a network port.

Scripts
Get data from any API, service, or database with a script.

Configure a new token for receiving data over HTTP. Learn More

Name: WLANPi HEC Token

Source name override ?: optional

Description ?: optional

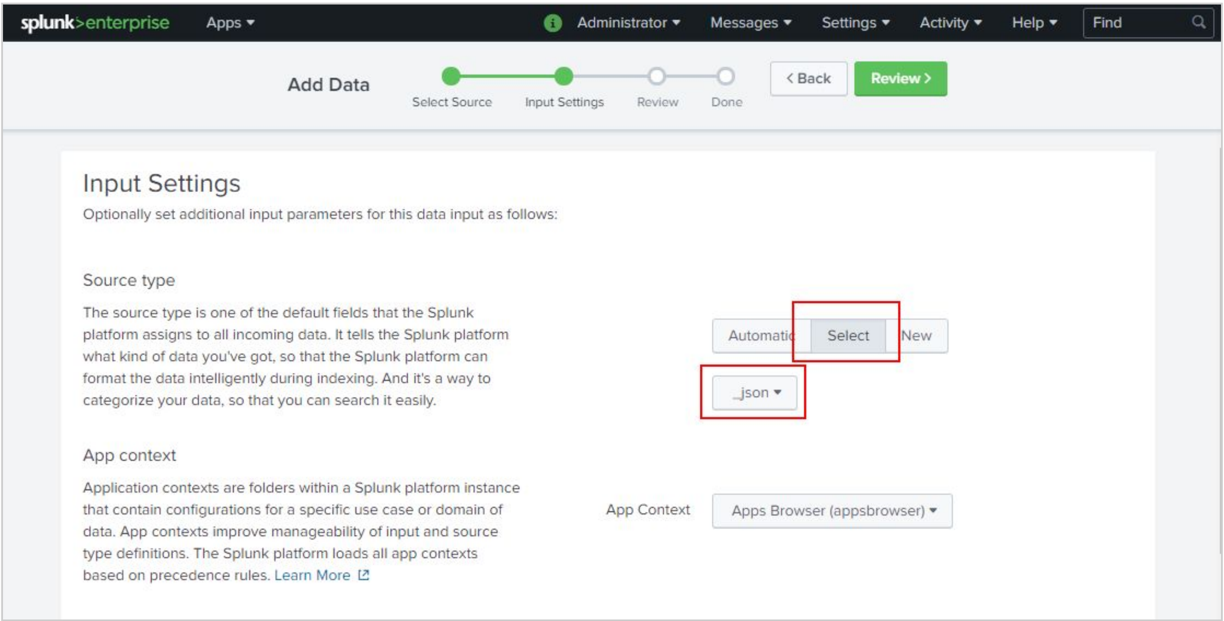
Output Group (optional): None ▾

Enable indexer acknowledgement ☐

FAQ

- What is the HTTP Event Collector?
- How do I set up the HTTP Event Collector?
- How do I view and configure the tokens that I can use to send data to the HTTP Event Collector?
- What clients can send data to the HTTP Event Collector?
- What port and protocol does the HTTP Event Collector receive data on and how can I change that?
- What is an output group?

In the next wizard panel select Source type: Select > Structured > _json :



Input Settings

Optionally set additional input parameters for this data input as follows:

Source type

The source type is one of the default fields that the Splunk platform assigns to all incoming data. It tells the Splunk platform what kind of data you've got, so that the Splunk platform can format the data intelligently during indexing. And it's a way to categorize your data, so that you can search it easily.

App context

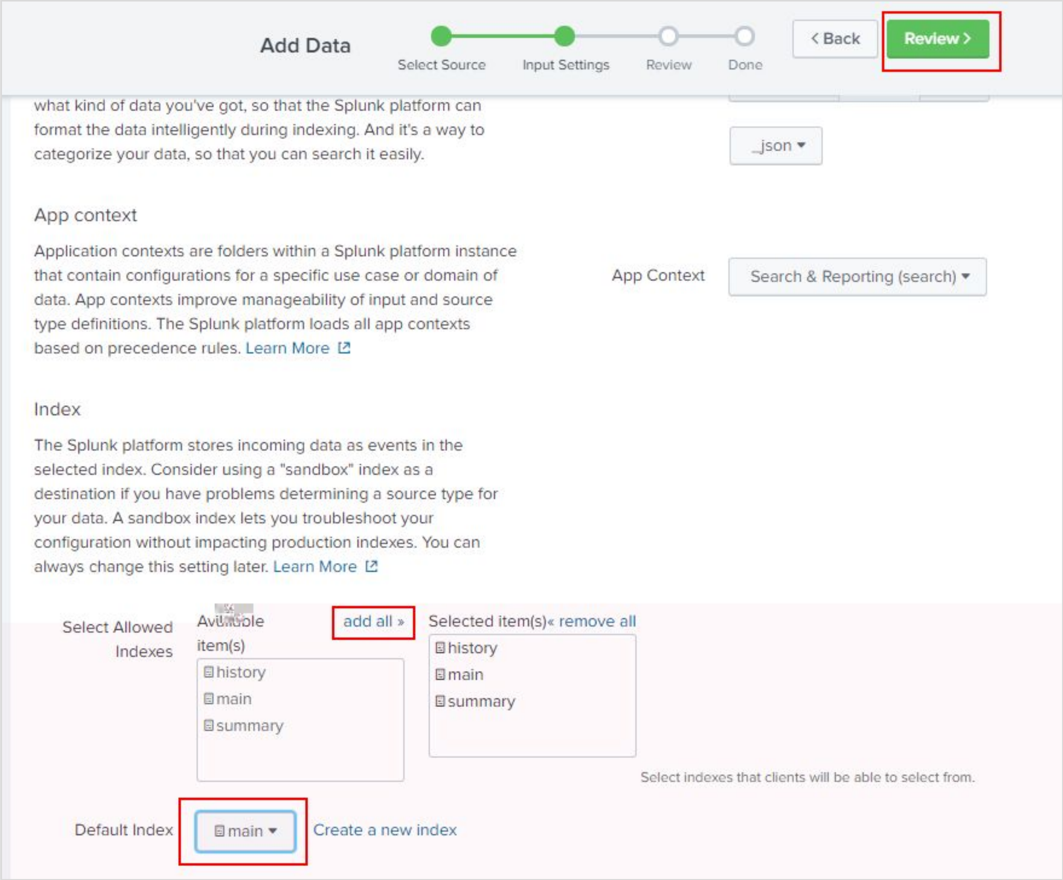
Application contexts are folders within a Splunk platform instance that contain configurations for a specific use case or domain of data. App contexts improve manageability of input and source type definitions. The Splunk platform loads all app contexts based on precedence rules. Learn More

App Context: Apps Browser (appsbrowser) ▾

Scroll down to the indexes and make the following selections:

- Select Allowed Indexes > add all
- Default Index : main

Next, hit the Review > button:



what kind of data you've got, so that the Splunk platform can format the data intelligently during indexing. And it's a way to categorize your data, so that you can search it easily.

App context

Application contexts are folders within a Splunk platform instance that contain configurations for a specific use case or domain of data. App contexts improve manageability of input and source type definitions. The Splunk platform loads all app contexts based on precedence rules. Learn More

App Context: Search & Reporting (search) ▾

Index

The Splunk platform stores incoming data as events in the selected index. Consider using a "sandbox" index as a destination if you have problems determining a source type for your data. A sandbox index lets you troubleshoot your configuration without impacting production indexes. You can always change this setting later. Learn More

Select Allowed Indexes

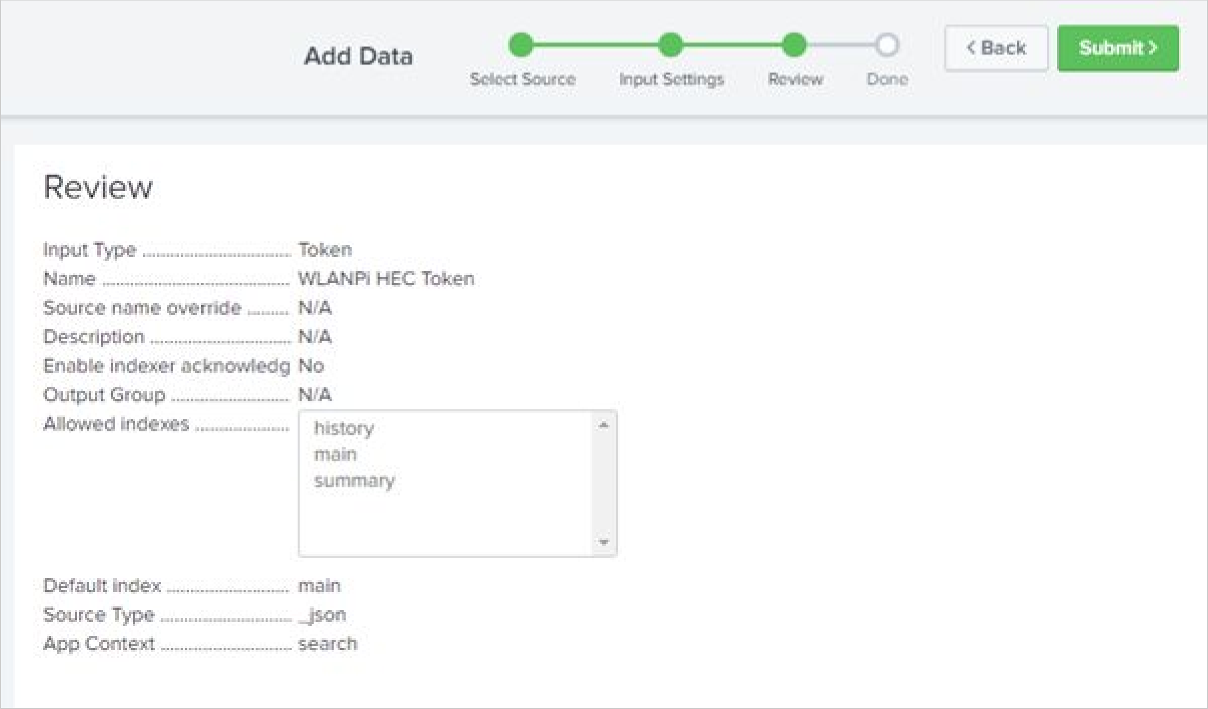
Available item(s): history, main, summary

Selected item(s): history, main, summary

Select indexes that clients will be able to select from.

Default Index: main ▾

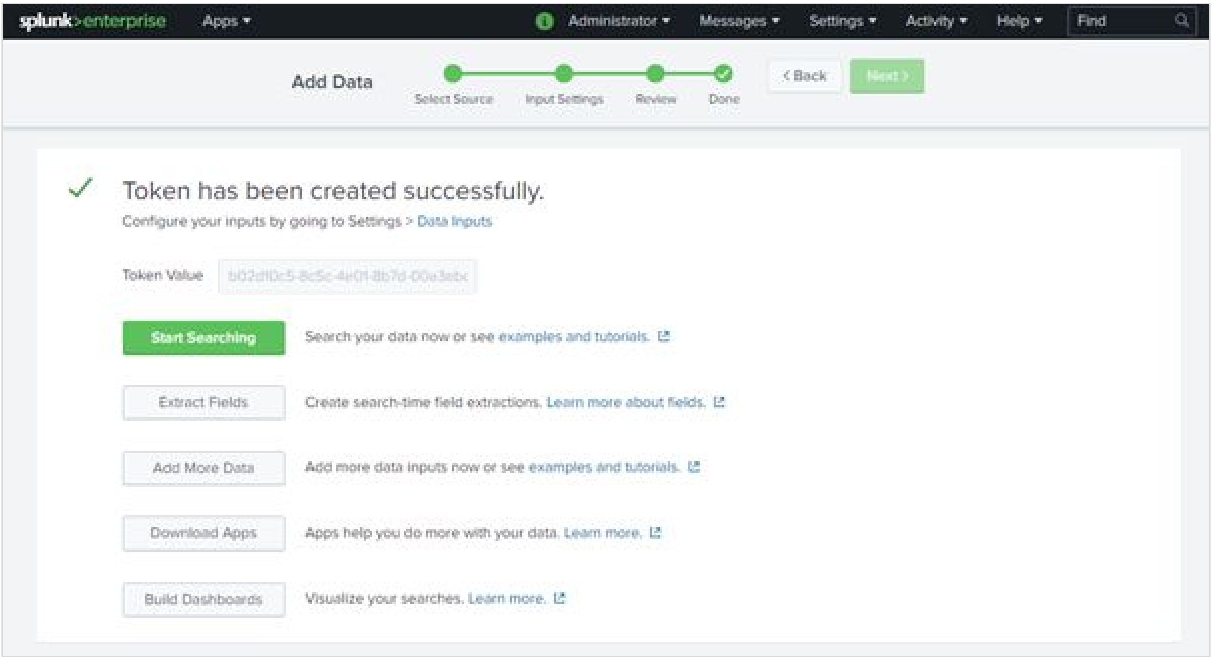
The token review panel is now should and should look like the graphic below. Finally hit the Submit > button:



Review

Input Type .... Token
Name .... WLANPi HEC Token
Source name override .... N/A
Description .... N/A
Enable indexer acknowledg No
Output Group .... N/A
Allowed indexes .... history
main
summary
Default index .... main
Source Type .... _json
App Context .... search

A final confirmation message will be provided as shown below:



✓ Token has been created successfully.

Configure your inputs by going to Settings > Data Inputs

Token Value b02d10c5-8c5c-4e01-8b7d-00a3ebc

Search your data now or see examples and tutorials.

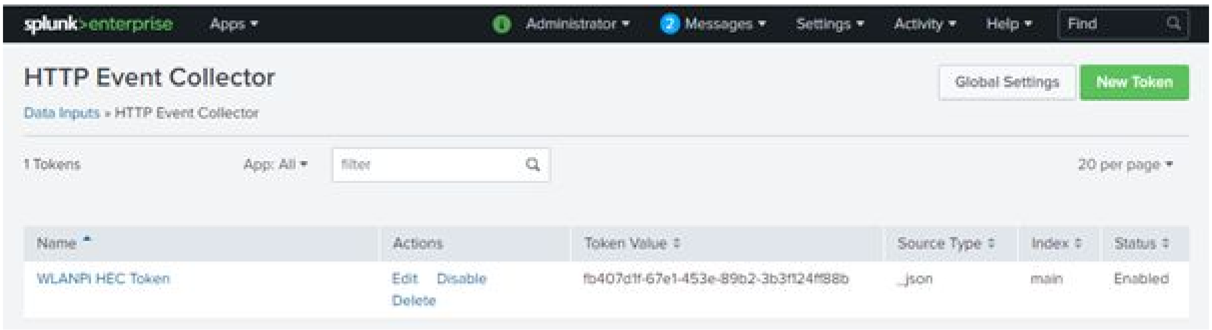
Create search-time field extractions. Learn more about fields.

Add more data inputs now or see examples and tutorials.

Apps help you do more with your data. Learn more.

Visualize your searches. Learn more.

If you return to Settings > Data Input > HTTPS Event Collector, you will now see the token your WLANPi will need to communicate with the Splunk server:



HTTP Event Collector

Data Inputs » HTTP Event Collector

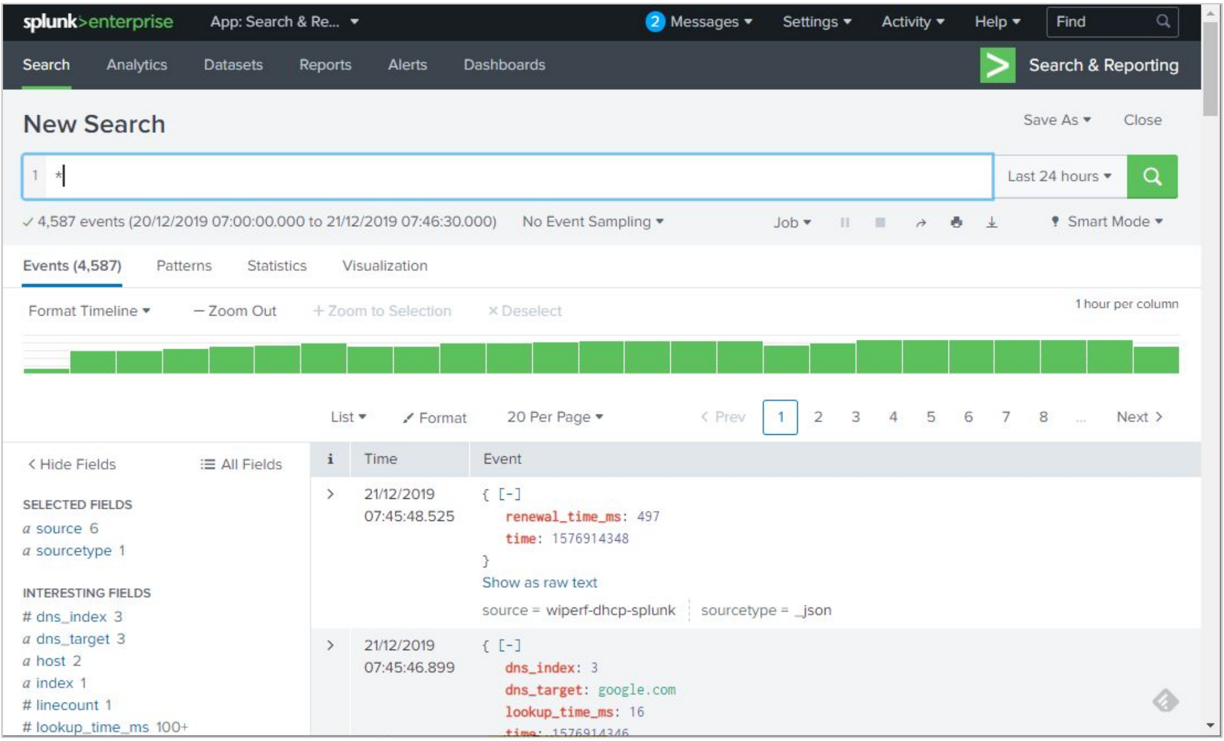
1 Tokens

Name ▴	Actions	Token Value ⇕	Source Type ⇕	Index ⇕	Status ⇕
WLANPi HEC Token	Edit Disable Delete	fb407d1f-67e1-453e-89b2-3b3f124ff88b	_json	main	Enabled

At this point, the Splunk server is ready to receive data from the WLANPi. Ensure that your WANPi is in Wiperf mode and has been configured with the correct server IP address, port number and the token we have just created above (copy and paste the “Token Value” in to your WLANPi config.ini file).

5.1.4 Perform a Test Search

After a few minutes, when the WLANPi has run a test cycle, data should start to appear in Splunk. The quickest way to check is to do a general search for data in Splunk and see what is being received. Go to “Apps : Search & Reporting > Search & Reporting” (top menu bar) and enter a “*” in the “New Search” text box. Results data should be seen as shown below:



1 *

✓ 4,587 events (20/12/2019 07:00:00.000 to 21/12/2019 07:46:30.000) No Event Sampling

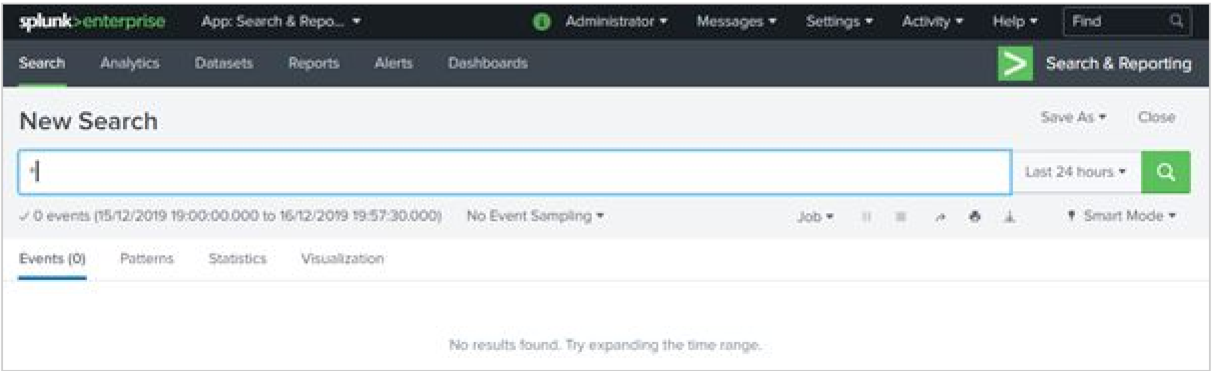
SELECTED FIELDS
a source 6
a sourcetype 1

INTERESTING FIELDS
# dns_index 3
a dns_target 3
a host 2
a index 1
# linecount 1
# lookup_time_ms 100+

```
> 21/12/2019      { [-]
  07:45:48.525       renewal_time_ms: 497
                     time: 1576914348
                  }
                  Show as raw text
                  source = wiperf-dhcp-splunk   sourcetype = _json

> 21/12/2019      { [-]
  07:45:46.899       dns_index: 3
                     dns_target: google.com
                     lookup_time_ms: 16
                     time: 1576914346
```

If your search result looks like this (no results found message), then you need to wait a little longer for data to arrive, or there is likely a comms problem between your WLANPi and Splunk:



*

✓ 0 events (15/12/2019 19:00:00.000 to 16/12/2019 19:57:30.000) No Event Sampling

No results found. Try expanding the time range.

5.2 Create a Dashboard

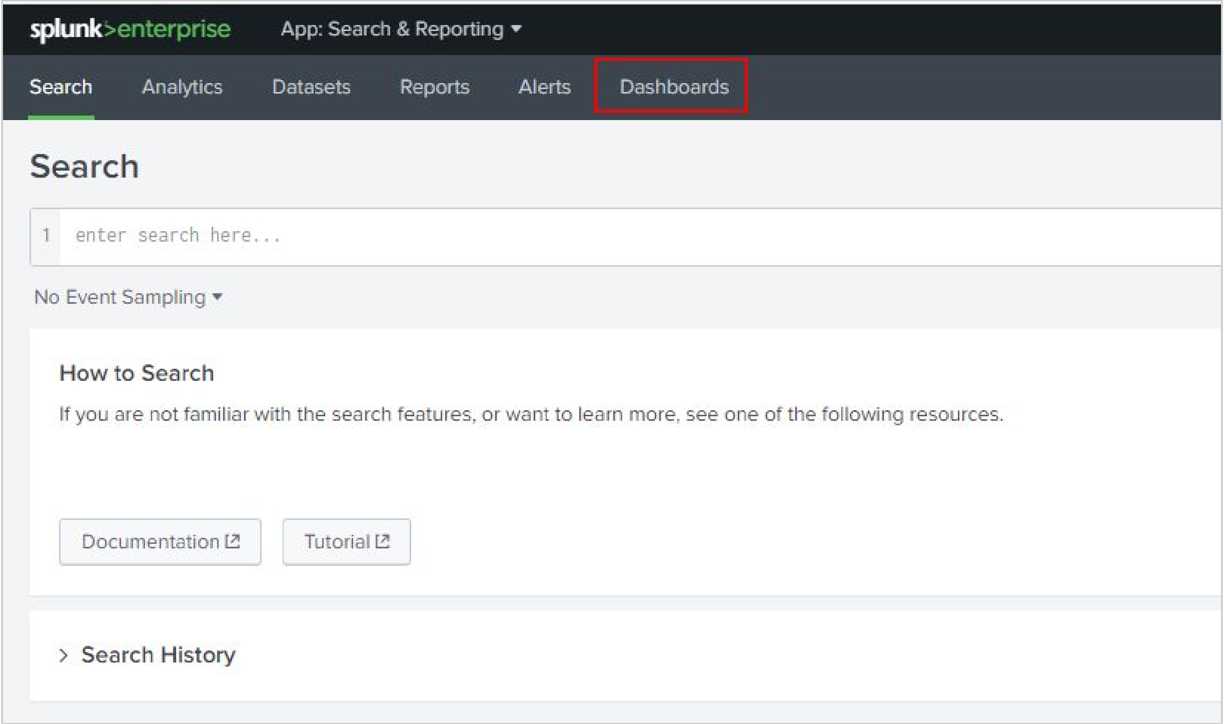
Now that we have data arriving at our Splunk server, we need to view the data in an interesting format. Splunk allows us to create a number of dashboards to visualize our data. We will now create a simple dashboard to demonstrate the visualization capabilities.

In the WLANPi's /home/wlanp/wiperf/dashboards directory, a number of pre-canned dashboard files have been provided to allow a dashboard to be copied and pasted easily. These are also available on the GitHub page of the Wiperf project:

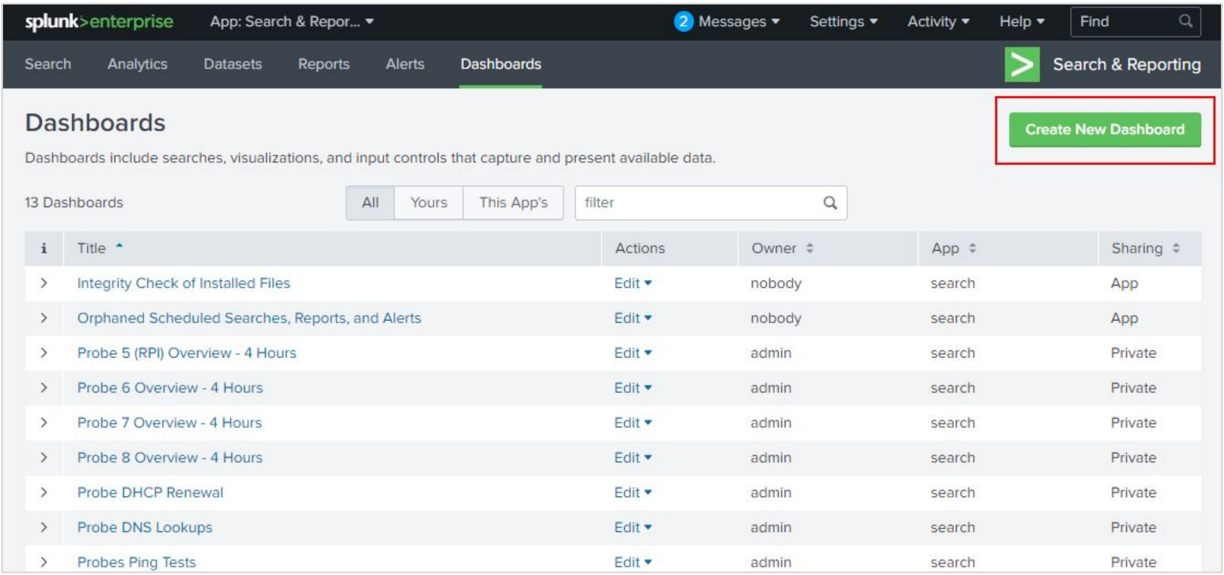
- https://github.com/wifinigel/wiperf/tree/master/dashboards

Use an SFTP client to pull the "probe_summary.xml" file from your WLANPi or open the file on the GitHub page and select "Raw" to copy and paste the code in to a local file on your laptop.

In the Splunk GUI, go to "Apps : Search & Reporting > Search & Reporting" (top menu bar) and hit the "Dashboards" link:



Hit the Create New Dashboard button:



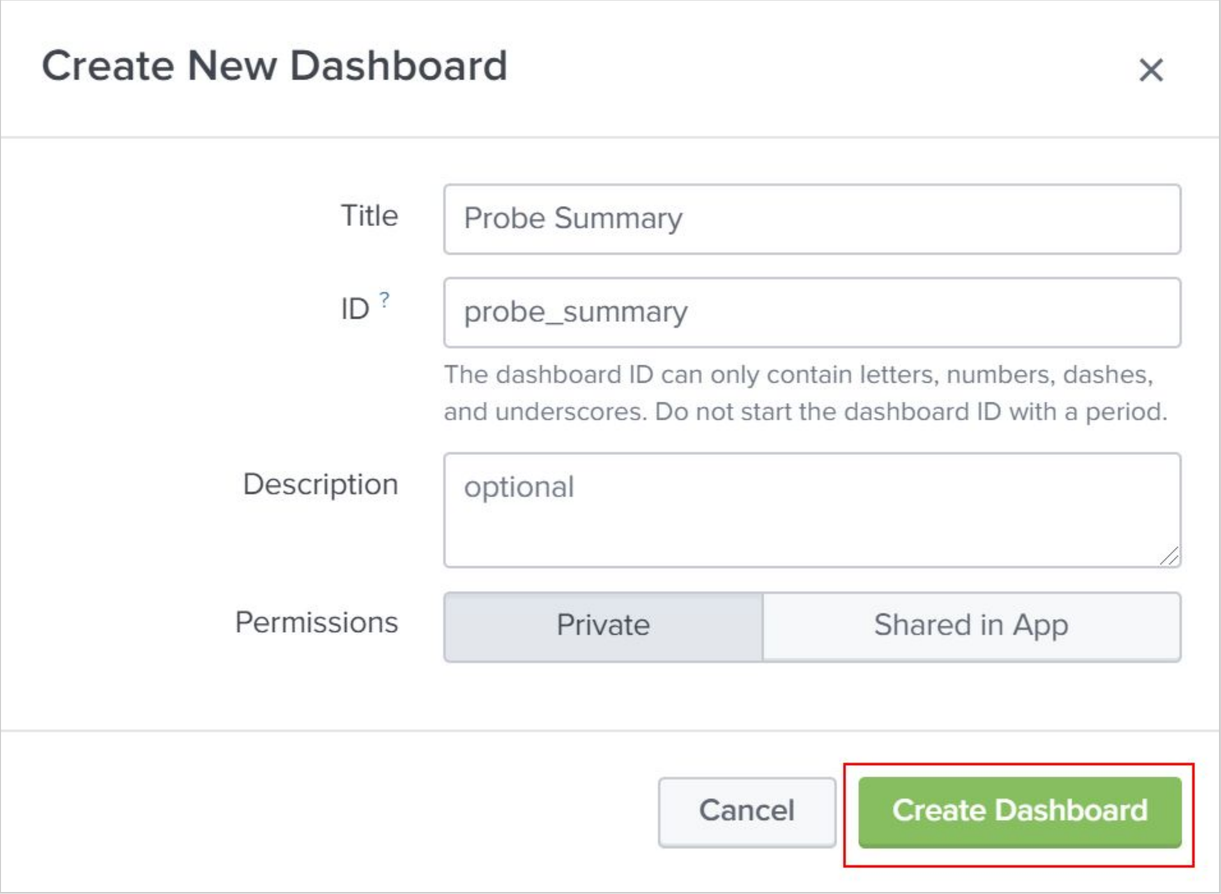
Dashboards

Dashboards include searches, visualizations, and input controls that capture and present available data.

13 Dashboards

i	Title ▴	Actions	Owner ⇕	App ⇕	Sharing ⇕
>	Integrity Check of Installed Files	Edit ▾	nobody	search	App
>	Orphaned Scheduled Searches, Reports, and Alerts	Edit ▾	nobody	search	App
>	Probe 5 (RPI) Overview - 4 Hours	Edit ▾	admin	search	Private
>	Probe 6 Overview - 4 Hours	Edit ▾	admin	search	Private
>	Probe 7 Overview - 4 Hours	Edit ▾	admin	search	Private
>	Probe 8 Overview - 4 Hours	Edit ▾	admin	search	Private
>	Probe DHCP Renewal	Edit ▾	admin	search	Private
>	Probe DNS Lookups	Edit ▾	admin	search	Private
>	Probes Ping Tests	Edit ▾	admin	search	Private

In the pop-up panel, enter a dashboard name and hit the Create Dashboard button:



Title: Probe Summary

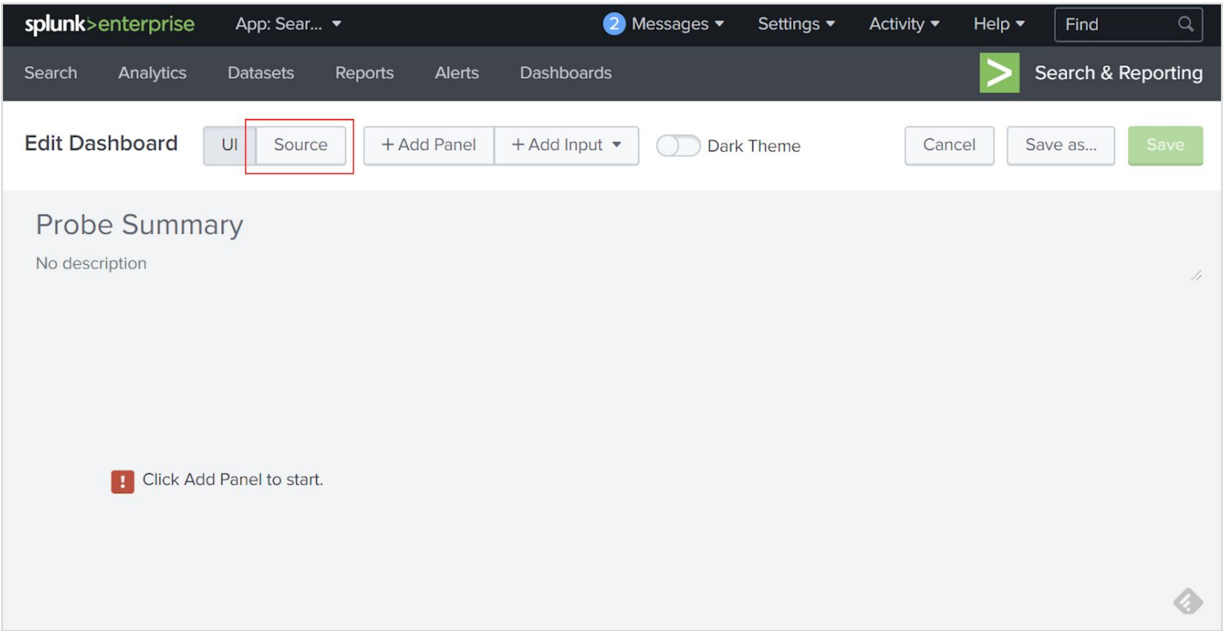
ID ?: probe_summary

The dashboard ID can only contain letters, numbers, dashes, and underscores. Do not start the dashboard ID with a period.

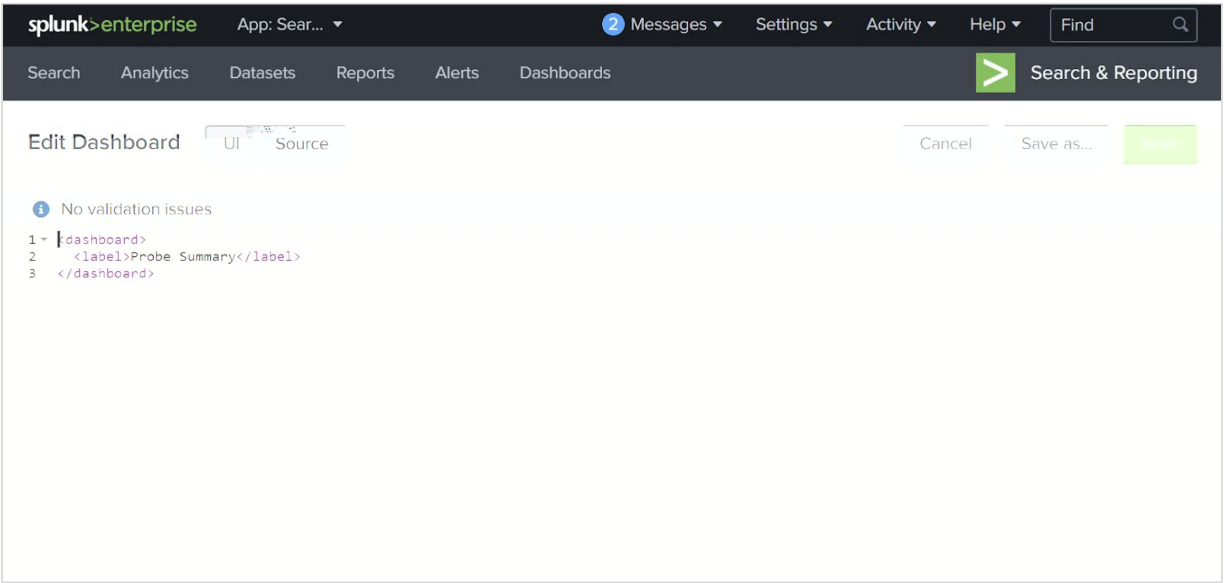
Description: optional

Permissions: [✓ Private] Shared in App

In the “Edit Dashboard” panel that opens, hit the “Source” button:



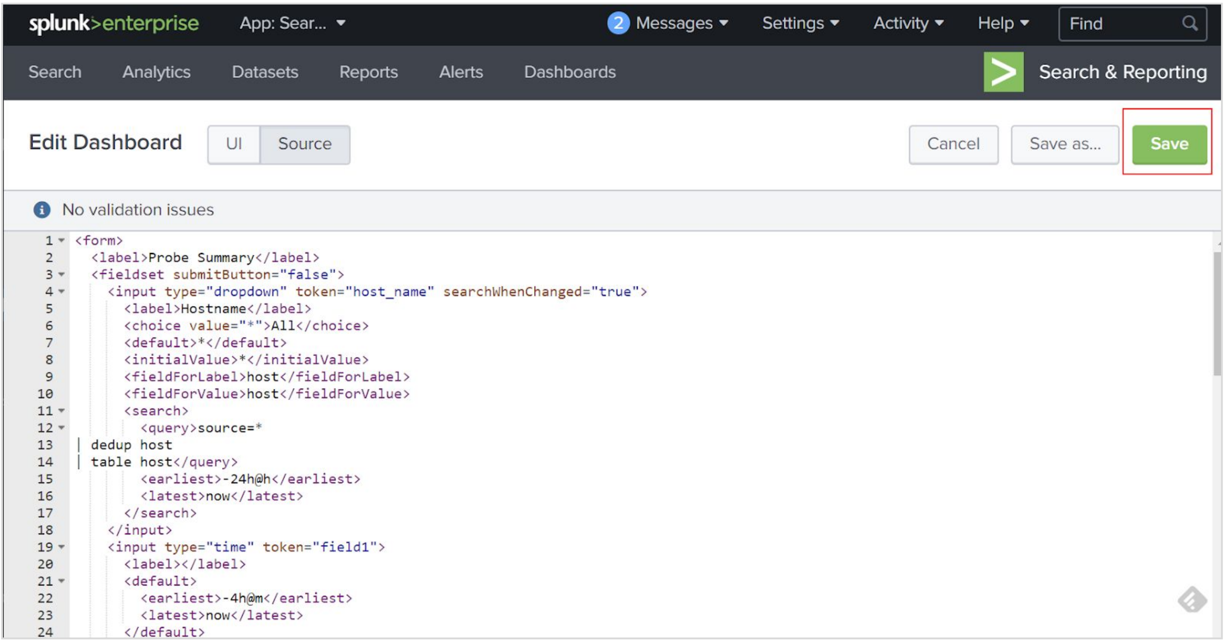
By default, some basic XML configuration will exist in the dashboard definition:



No validation issues

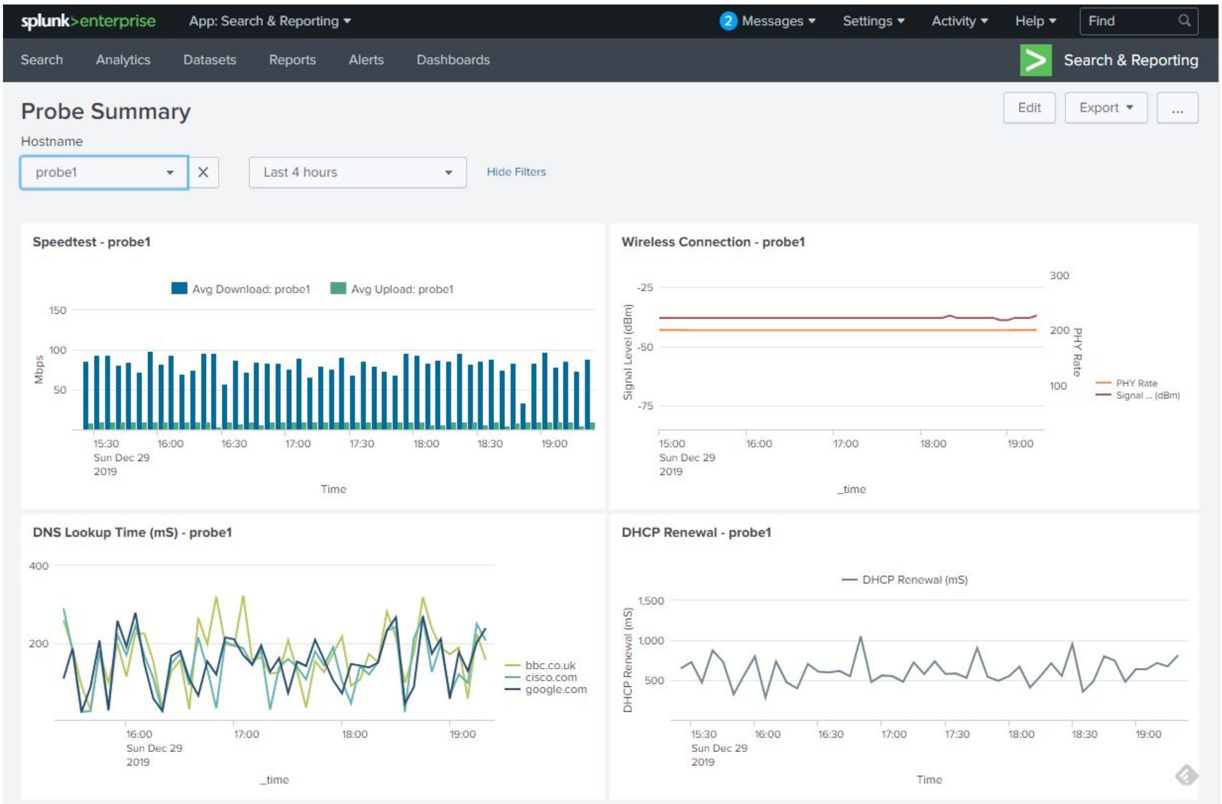
```
<dashboard>
  <label>Probe Summary</label>
</dashboard>
```

Open up the dashboard definition file previously downloaded from your WLANPi or the GitHub site in a text editor. Then simply paste in the code as shown below (make sure the original code was all removed):



```
<form>
  <label>Probe Summary</label>
  <fieldset submitButton="false">
    <input type="dropdown" token="host_name" searchWhenChanged="true">
      <label>Hostname</label>
      <choice value="*">All</choice>
      <default>*</default>
      <initialValue>*</initialValue>
      <fieldForLabel>host</fieldForLabel>
      <fieldForValue>host</fieldForValue>
      <search>
        <query>source=*
| dedup host
| table host</query>
        <earliest>-24h@h</earliest>
        <latest>now</latest>
      </search>
    </input>
    <input type="time" token="field1">
      <label></label>
      <default>
        <earliest>-4h@m</earliest>
        <latest>now</latest>
      </default>
```

After hitting the Save button, the dashboard will now be shown:



Using the hostname and time period selector above the graphs, different probes and reporting periods may be viewed.