Splunk Enterprise Data Administration
Lab Exercises

Table of Contents

Lab Environment Information

Lab typographical conventions

Replace following keys with the values indicated:

{student-ID}	Your assigned 2-digit student number
{os-user}	Your assigned OS account name (on your deployment/test server and forwarders)
{user-ID}	Your assigned Splunk username
{password}	Your assigned password
{DS-dns}	The domain name services hostname of your assigned deployment server
{DS-eip}	The external IP address of your assigned deployment server
{DS-iip}	The internal IP address of your assigned deployment server
{SH-dns}	The domain name services hostname of your assigned search head
{SH-eip}	The external IP address of your assigned search head

To support the lab activities, your lab environment also includes the following shared servers:

Hostname	IP address	Role
ip-10-0-0-50	**10.0.0.50**	Splunk universal forwarder #1 (UF1)
ip-10-0-0-100	**10.0.0.100**	Splunk universal forwarder #2 (UF2)
ip-10-0-0-77	**10.0.0.77**	Splunk Heavy Forwarder (HF)
ip-10-0-0-88	**10.0.0.88**	Splunk Indexer #1 (IDX1)
ip-10-0-0-99	**10.0.0.99**	Splunk Indexer #2 (IDX2)
ip-10-0-0-111	**10.0.0.111**	Splunk Search Head (SH)

The **SPLUNK_HOME** token indicates the directory where Splunk is installed on the host:

On Linux Indexer:	**/opt/splunk**
On Windows Indexer:	**C:\Program Files\Splunk**
On Heavy Forwarders:	**/opt/home/{os-user}/splunk**
On Universal Forwarders:	**/opt/home/{os-user}/splunkforwarder**

The following text editors are installed in your environment:

Linux server:	**nano**	
	vi	(If you are unfamiliar with **vi**, use **nano**. It is an easy text editor.)
Windows server:	**Notepad++**	

Some steps contain icons which denote the action to take on the appropriate OS.

	Linux OS
	Windows OS

NOTE: When you access the Splunk user interface for the first time, Splunk may ask if you want a tour of the app. Throughout the exercises, you can dismiss this prompt at any time.

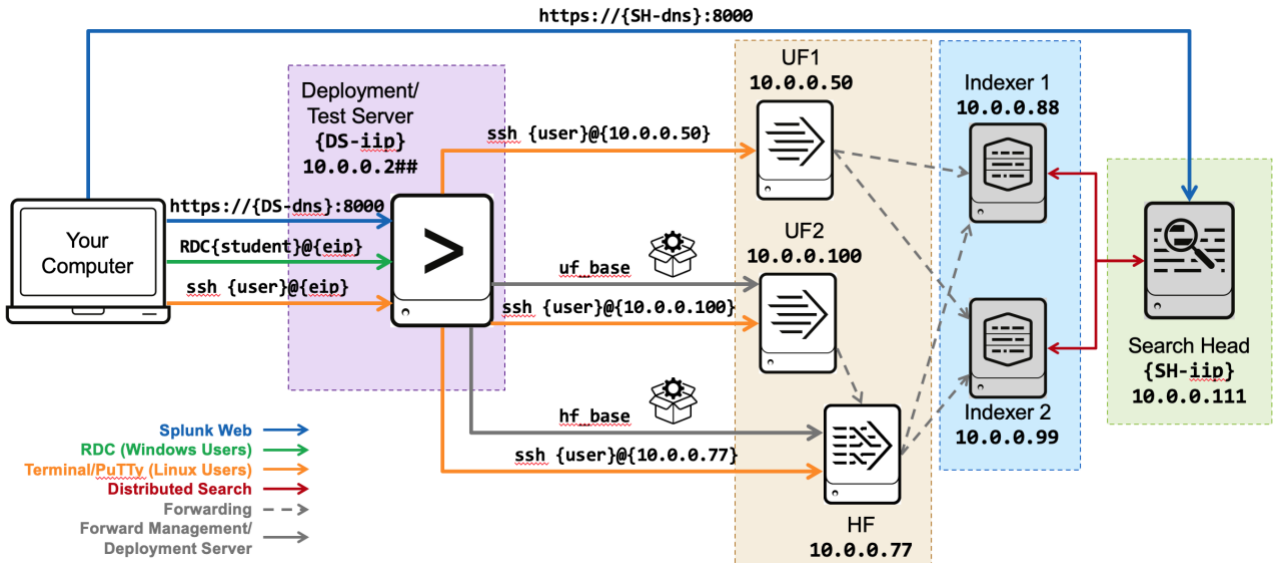
Lab Environment Overview

You will be working in a server environment. The diagram below provides an overview of this lab environment. You will be assigned user accounts, passwords, and an external IP address to access your deployment/test server and an external IP address to access the shared search head.

Command line access requires SSH/putty/RDC to the external IP address. You will SSH into the universal forwarders using the internal IP addresses. This can only be done after establishing an SSH/putty/RDC connection to your deployment/test server.

Depending on your lab environment configuration, your deployment/test server will either be a linux or Windows host running a Splunk Enterprise instance. All forwarders and the shared search head are configured on linux hosts.

Splunk Environment:



Splunk instance	Access
Search Head (search / verify data configs)	**power** role
Indexers	No access
Forwarders (data sources and inputs)	**admin** role
Deployment/Test Server	**admin** role

Module 1 Lab – Getting Data Into Splunk

Description

Welcome to the Splunk Enterprise Data Administration lab environment. In this exercise, you will perform basic configuration tasks using the Splunk Web interface and, using the CLI, investigate Splunk instance settings.

Please ensure you are able to identify all of the following values that have been provided to you.

Your student ID is a unique 2-digit identifier used throughout the lab exercises to differentiate your work from other class participants' work. When asked in the labs, substitute the "##" references with your student ID.

Student ID:	**{student-ID}**	*Example:* **03**

Search Head Credentials

This lab environment uses a shared search head. You will log into the search head using your unique assigned Splunk username, which has been assigned the Splunk power role. You will never log into the search head as **admin**.

Splunk Web URL:	**https://{SH-dns}:8000**	*Example:* **https://SH12345.students.splunk.education:8000**
Splunk Username	**{user-ID}**	*Example:* **alex**
Password:	**{password}**	*Example:* **splunk3du**

Deployment / Test Server Credentials

You have been assigned your own deployment / test server Splunk instance. Splunk Web (browser) access procedures are the same regardless of the underlying operating system. The command line access procedure is found below under **Host Operating System (OS) Credentials**, and depends upon the underlying OS.

Splunk Web URL:	**https://{DS-dns}:8000**	*Example:* **https://DS12345-03.students.splunk.education:8000**
Splunk Username	**admin**	*Example:* **admin**
Password:	**{password}**	*Example:* **splunk3du**

Host Operating System (OS) Credentials

To access the Linux filesystem, use an SSH client such as **Terminal** (Mac) or **PuTTY** (Windows).

To access the Windows filesystem, use a Remote Desktop client (RDC), such as Microsoft Remote Desktop.

Host domain name:	**{DS-dns}**	*Example:* **DS12345-03.students.splunk.education**
Host IP address:	**{DS-eip}**	*Example:* **52.5.196.118**
Linux username:	**{os-user}**	*Example:* **alex**
Windows username:	**student**	*Example:* **student**
Password:	**{password}**	*Example:* **splunk3du**

Steps

You will access the shared search head **{SH-dns}** and your personal deployment/test server **{DS-dns}** instances frequently with Splunk Web throughout the lab exercises.

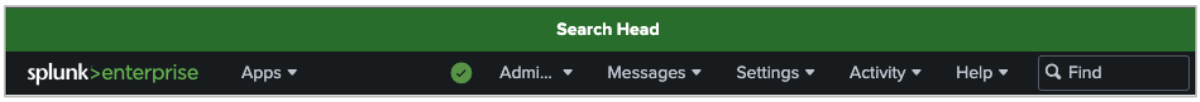
In this first lab you will change the color of the Global Banner in Splunk Web so that you can more easily distinguish that you are on the correct instance in later labs.

Additionally you can use one of the following options to context-switch easily between them when necessary:

- Option 1: Keep a separate tab or window open to each machine. If you're not sure which instance you are currently accessing, click the **Settings** menu. If you see an abridged list of options, you're on the search head. If you see a full list of options, you're on your deployment/test server.
- Option 2: Use two different web browsers. For example, use Chrome to access your search head and Firefox to access the deployment test server.

Task 1: Access Splunk Web on the Search Head.

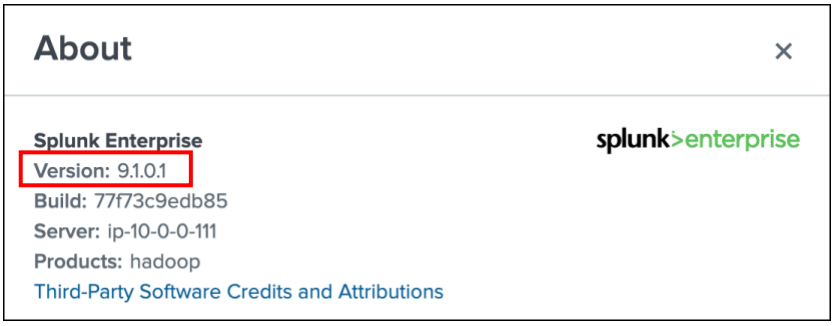
1. Navigate to the search head (using your browser of choice): **https://{SH-dns}:8000**.
2. Log in with your assigned **{user-ID}** and password **{password}**.
3. Verify that a green banner appears above the Splunk Web navigation bar.



A green banner stating "Search Head" has been configured in the **Server Settings > Global Banner** settings to help you easily identify when you are logged into Splunk Web on the Search Head.

You will configure a similar banner for your Deployment/Test server later in this lab.

4. From the Splunk bar, to identify the Splunk version that the search head is running, click **Help > About**.



5. From the Splunk bar, click your **SH_{user-ID}** name, and click **Account Settings**.

   Notice in the **Full name** field your assigned **SH_{user-ID}**. **Do not change**.

   The **Email address** field contains a two-digit number. This is your **{student-ID}** (leading zero required for student IDs 01-09). **Do not change**.

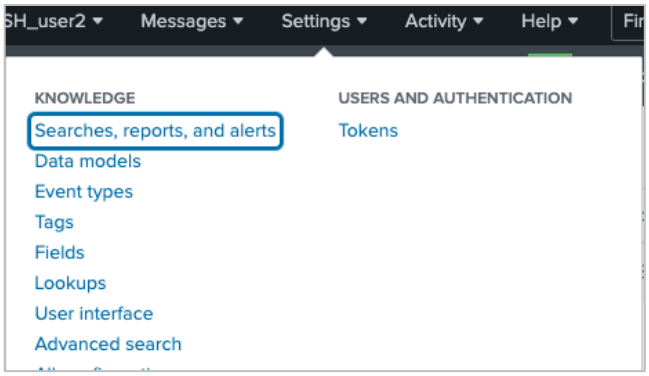
NOTE: Do not change your assigned password.

6. From the Splunk bar, click your **SH_{user-ID}** name and click **Preferences**.

   The **Global** setting should be selected

7. In the **Default** application field, select **Search & Reporting**.
8. Click **Apply**.
9. In the app navigation bar, click **Apps > Search & Reporting**.
10. Click **Skip/Skip tour** to dismiss any tour messages.
11. Click **Settings**.

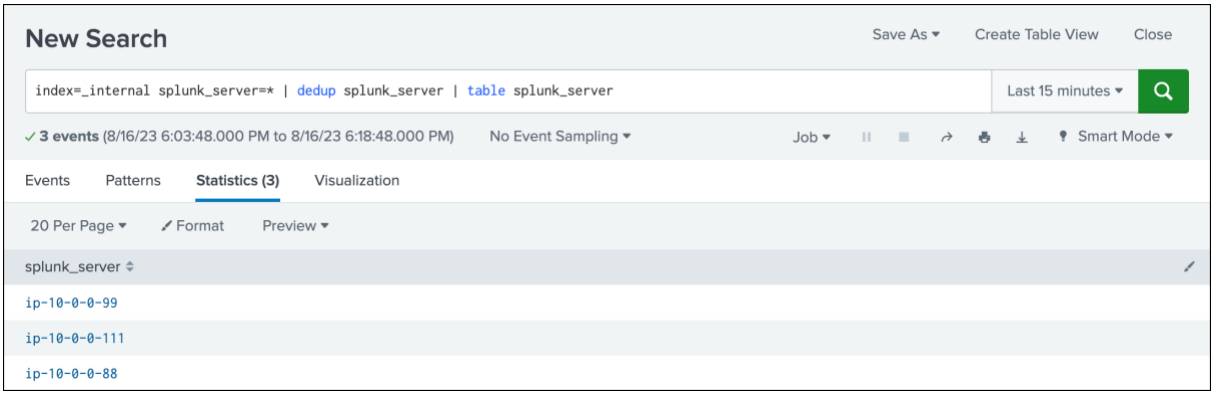
    The options shown are the defaults available to the Splunk **power** role:



Task 2: Run a search on the Search Head.

12. Navigate to the Search & Reporting app by clicking on the **splunk>enterprise** logo in the top left of Splunk Web
13. To Identify some of the Splunk components in your environment, execute the following search over the **Last 15 minutes**:

```
index=_internal splunk_server=* | dedup splunk_server | table splunk_server
```



NOTE: On a standard default server, power users cannot search the **_internal** index. This was modified in the training environment.

The table lists the Splunk servers that are currently searchable by the search head:

Hostname	IP address	Role
`ip-10-0-0-88`	`10.0.0.88`	Splunk Indexer #1 (IDX1)
`ip-10-0-0-99`	`10.0.0.99`	Splunk Indexer #2 (IDX2)
`ip-10-0-0-111`	`10.0.0.111`	Splunk Search Head (SH)

NOTE: If you see more servers in your data table, it indicates other class participants have already completed subsequent lab exercises.

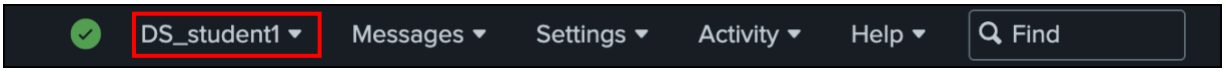
Task 3: Use Splunk Web on the deployment/test server to change server settings.

14. Open a separate tab or window in your browser and navigate to your deployment/test server instance:

    `https://{DS-dns}:8000`

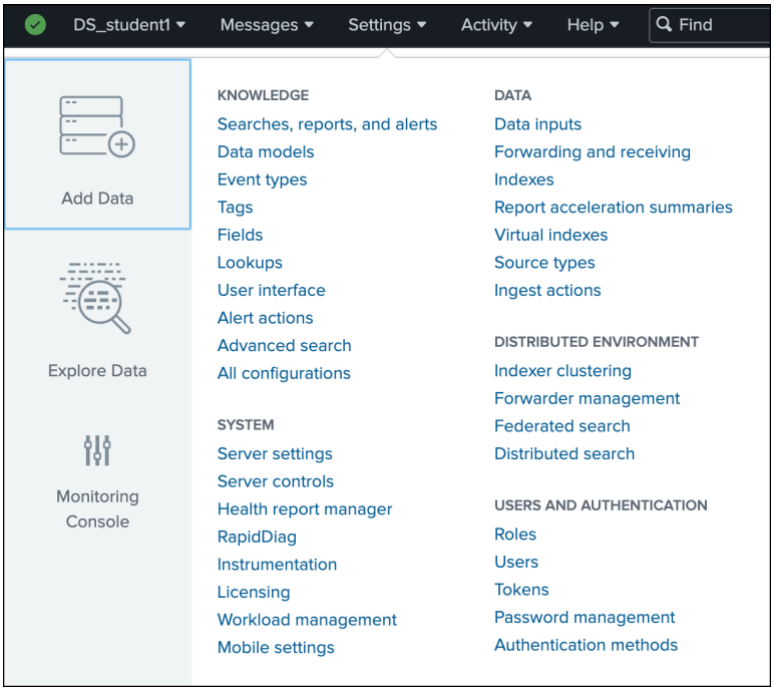
15. Log in as **admin** using your assigned password {`password`}.
16. Dismiss any unnecessary informational messages.
    - Click **Got it!** in the **"Helping You Get More Value from Splunk Software"** pop-up page.
17. Verify your assigned user ID in the Splunk bar with a **DS**_ prefix.

    This prefix identifies your login session and the deployment/test server.



18. Click **Settings**.

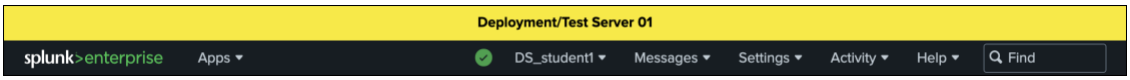
    The full list of options is displayed for this role. You are assigned the **admin** role with full administrator privileges on this Splunk instance.



19. Navigate to **Settings > Server Settings > Global banner**.
20. On the **Customize global banner** page, set the following values **(**where ## is your **{student-ID})**:

Banner Visibility	**On**
Background Color	**Yellow**
Message	**Deployment/Test Server ##**

21. Click **Save > Ok**.
22. Verify that a yellow banner now appears above the Splunk Web navigation bar.



23. Click **DS_{user-ID} > Preferences**.

    The **Global** setting should be selected

24. Change the **Time zone** to your current time zone, then click **Apply**.
25. Click **DS_{user-ID} > Account Settings**.

    Notice in the **Full name** field your assigned **DS_{user-ID}**. **Do not change**.

    In the **Email address** field, notice your two-digit **{student-ID}**. Leading zero required for student IDs 01-09. **Do not change your value**.

26. Navigate to **Settings > Server settings > General settings**.

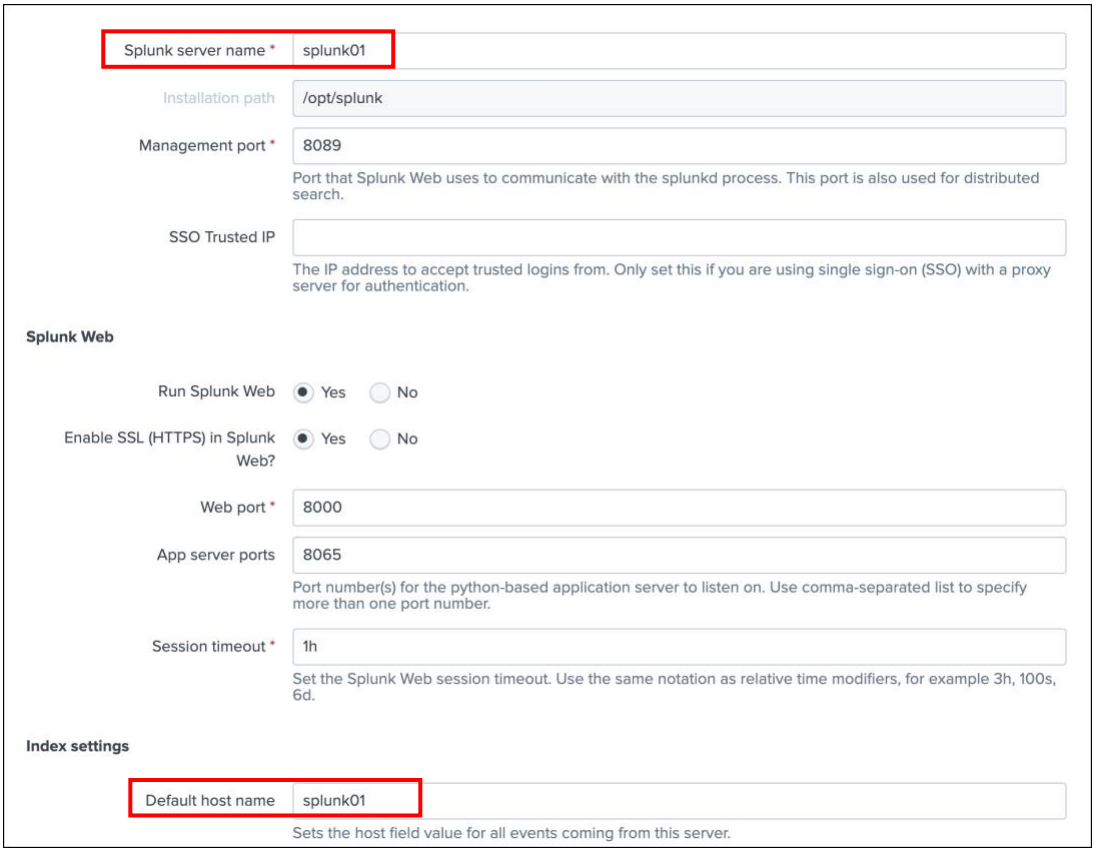
    Make note of the path specified in the **Installation path** field:

	/opt/splunk
	C:\Program Files\Splunk

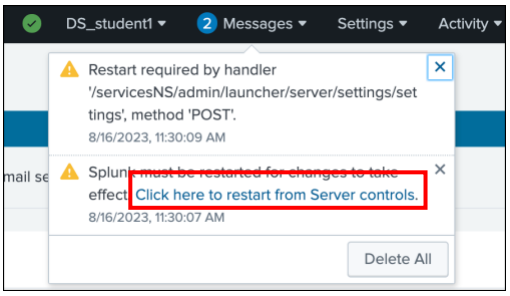
    This directory where Splunk is installed is referred to as **SPLUNK_HOME**.

27. Rename the Splunk server name and default host name:

Splunk server name:	**splunk##**	where ## is your **{student-ID}**
Default host name:	**splunk##**	where ## is your **{student-ID}**



28. Click **Save**. (These changes require a restart of Splunk.)
29. Click **Messages > Click here to restart from Server controls**.



30. In the resulting dialog box,click **Restart Splunk > OK**.
31. Click **OK** when the dialog box indicates that the restart was successful.
32. After the restart, log back into Splunk Web with user **`admin`** and your assigned password.

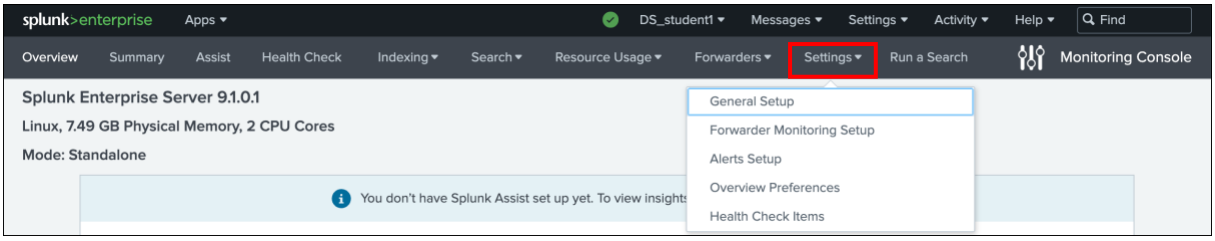
    After logging in, you should see the home screen of Splunk Web. If you instead see the **Server controls** page do *not* click the **Restart Splunk** button again.

Task 4: Configure the Monitoring Console (MC) on the deployment/test server.

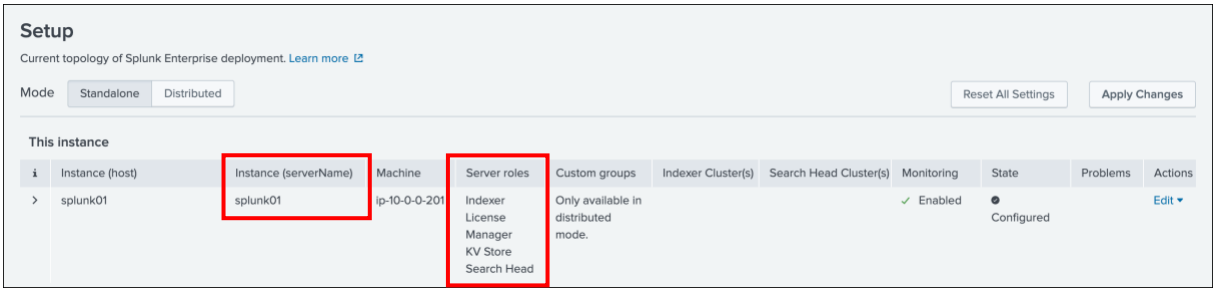
33. In Splunk Web on the deployment server, navigate to **Settings > Monitoring Console**.

    Look for the **Monitoring Console** icon on the left side of the **Settings** menu.

34. On the Monitoring Console navigation bar (the dark grey bar found under the black Splunk Web navigation bar) click **Settings** > **General Setup**.



35. Verify the server name and make a note of the discovered server roles.



36. Click **Edit > Edit Server Roles**.

NOTE: You may need to scroll to the right in the table to see the **Edit** hyperlink, depending on the size of your browser window.

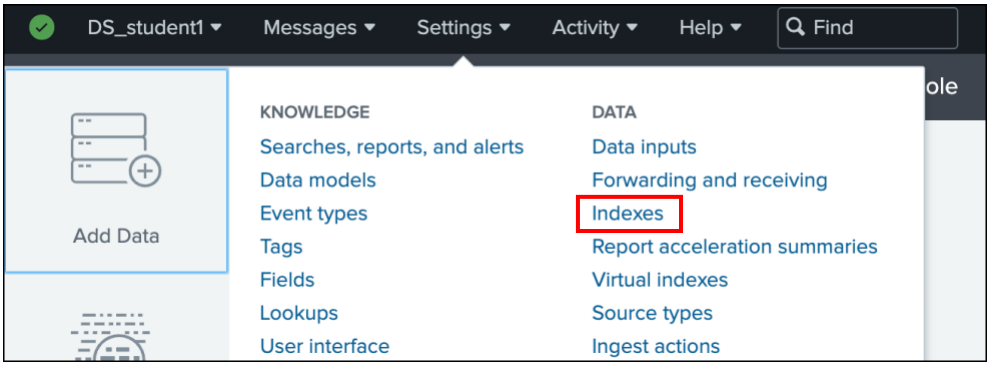
37. Remove the check mark from **Search Head**, and select the check mark for **Deployment Server**, then click **Save > Done**.
38. To complete the app setup, click **Apply Changes > Go to Overview**.
39. On the **Overview** page, review the following:
    - Monitoring Console is running in standalone mode.

    - No errors are displayed.
    - No excessive resource usage is detected. The CPU Usage and Memory Usage rates should be low (less than 20%).

Task 5: Create a local test index on the deployment/test server.

You create a **test** index to ingest data into. You learn about indexes in a later module.

40. In Splunk Web on the deployment server, navigate to **Settings > Indexes**.

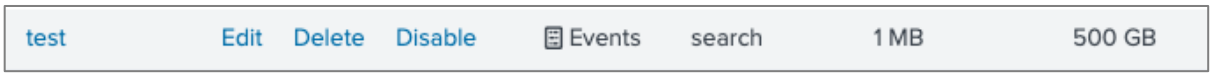


41. Click **New Index**.
42. Populate the form as follows:

    Index Name: **`test`**
    App: **`Search & Reporting`**
    (This saves the configurations within the **`search`** app-context).

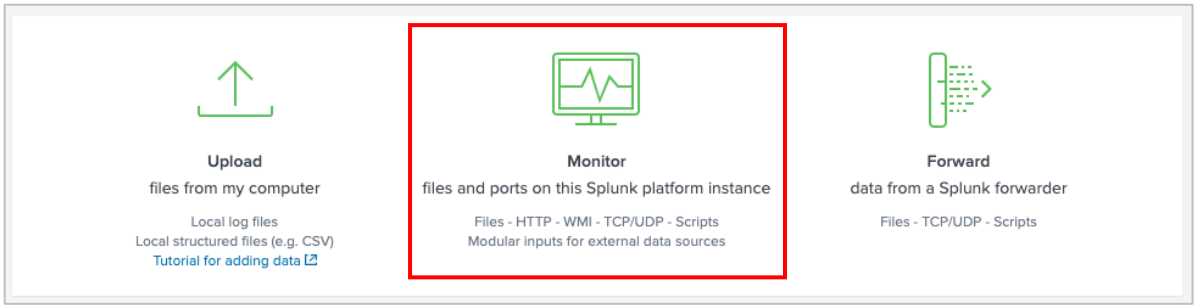
    Notice the default **Index Data Type** is **`Events`**. Leave the rest of the fields empty to accept defaults.

43. Click **Save**.
44. Verify the **`test`** index now appears at the bottom of the **Indexes** list.



Task 6: Index events from an access.log file to a test index.

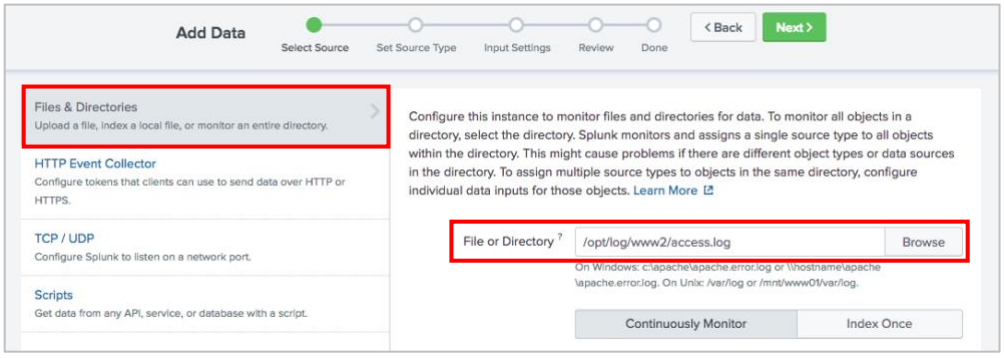
45. Click **Settings > Add Data**.
46. If you see the **Welcome**, ***username*** "Would you like to take a quick tour?" message, click **Skip**.
47. Click **Monitor** to launch the **Add Data** wizard.



48. On the **Select Source** step, click **Files & Directories**.

49. Click **Browse**, navigate to the file listed below, click the file name (**access.log**), then click **Select**:

	`/opt/log/www2/access.log`
	`C:\opt\log\www2\access.log`

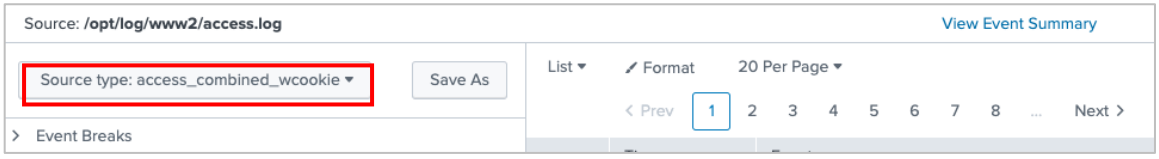


50. Make sure **Continuously Monitor** is selected.

NOTE: This selection creates a monitor stanza in the **inputs.conf** file. (The **inputs.conf** is created if it does not already exist.) You examine the **inputs.conf** file in a later module.

51. Click **Next** to go to the **Set Source Type** page.

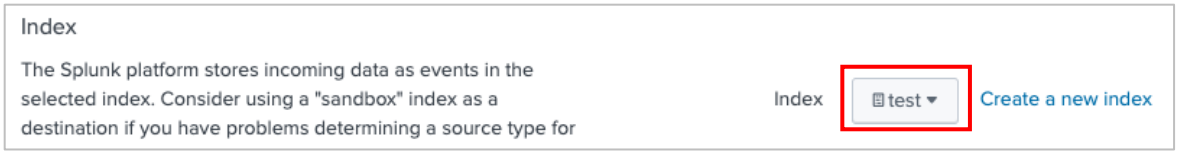
NOTE: Splunk auto-selected the **access_combined_wcookie** source type. This will be discussed later.



52. Click **Next** again to go to the **Input Settings** page and confirm the following selection:

App Context: **`Search & Reporting`**
Host field value: **`splunk##`** (The ## should be your Student ID number)

53. For **Index**, select **`test`**.



54. Click **Review**. The summary of the input should look as follows:

Input Type	**File Monitor**
Source Path	**C:\opt\log\www2\access.log** (Windows server) **/opt/log/www2/access.log** (Linux server)
Continuously Monitor	**Yes**
Source Type	**access_combined_wcookie**
App Context	**search**
Host	**splunk##**
Index	**test**

55. Click **Submit**.

Check Your Work

Task 7: Confirm your input configuration.

56. To verify your monitor input, click **Start Searching**.
57. Observe the search string:

```
source="/opt/log/www2/access.log" host="splunk##" index="test"
   sourcetype="access_combined_wcookie"
```

```
source="C:\\opt\\log\\www2\\access.log" host="splunk##" index="test"
   sourcetype="access_combined_wcookie"
```

NOTE: Splunk Search Processing Language (SPL) regular expressions are PCRE (Perl Compatible Regular Expressions). The backslash character (****) is used in regular expressions to "escape" special characters.

On Windows you will notice pairs of backslashes (****) in the SPL search string. This is because the backslash needs to be escaped. For more details, refer to **Backslash characters** at: https://docs.splunk.com/Documentation/Splunk/latest/Search/SPLandregularexpressions

58. Observe the automatically extracted field names and values:

Note the following fields and values:

source	**`C:\opt\log\www2\access.log`** (Windows server) **`/opt/log/www2/access.log`** (Linux server)
sourcetype	**`access_combined_wcookie`**
host	**`splunk##`**

Troubleshooting Suggestions

NOTE: The following steps require access to the Splunk server command terminal. Follow the steps shown in Lab exercise 2, Task 1 to access the command terminal of your deployment/test server.

1. If you can't access Splunk Web, make sure the Splunk service is running. In the terminal, run:

```
./splunk status
```

```
splunk status
```

2. If **`splunkd`** is not already running, start the **`splunkd`** service.

```
./splunk start
```

```
splunk start
```

Module 2 Lab – Config Files and Apps

Description

In this lab exercise, you will connect to your deployment/test server and view some Splunk configuration files, and use some Splunk commands to validate your configuration on disk and in memory.

Steps

Task 1: Access the command terminal of your deployment/test server.

1. Connect to the command line of your dedicated Splunk deployment/test server.

If your Splunk lab server is Linux, use one of these two methods:

1. If your personal computer is running *nix or macOS, start an SSH session to your deployment/test server by opening a terminal window and executing:

```
ssh {os-user}@{DS-eip}
```

When prompted for the authenticity of the host and the key fingerprint, type “yes”.

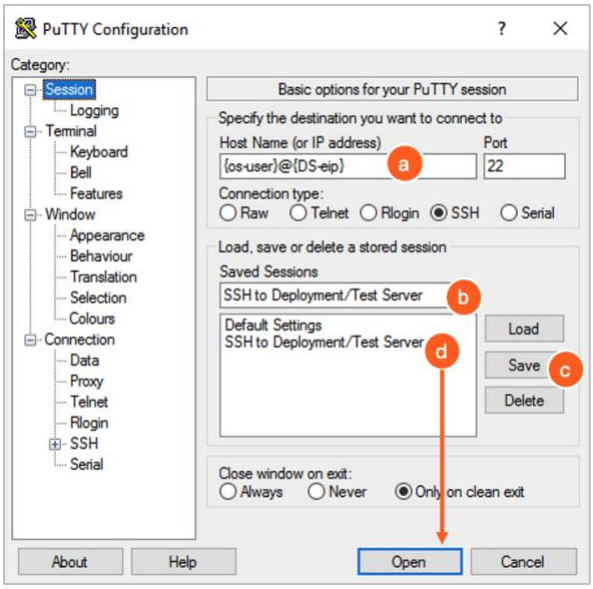
2. OR, if your personal computer is Windows, use an SSH client, such as PuTTy. (PuTTY is a free and reliable SSH client found at https://www.putty.org/). To use PuTTy to start an SSH session to your deployment/test server:

   a. Replace **{os-user}@{DS-eip}** with your designated values.

   b. Name your session, for example “**SSH to Deployment/Test Server**”

   c. Save the session for later re-use.

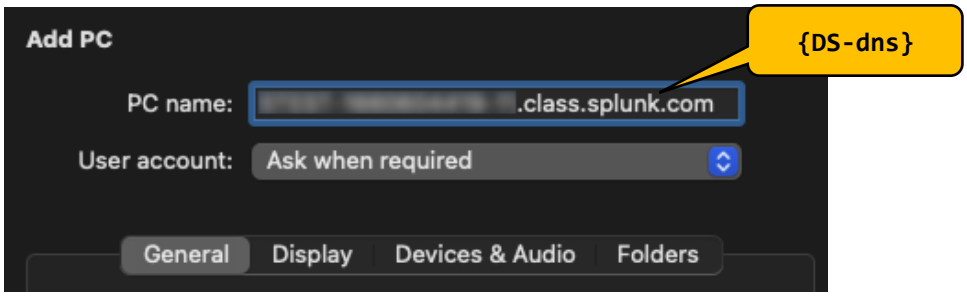
   d. Click on the session “**SSH to Deployment/Test Server**” and click **Open**.



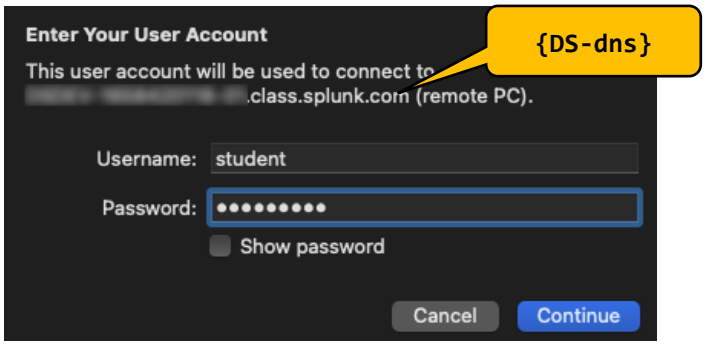
   e. When prompted for the authenticity of the host and the key fingerprint, type “yes” to continue.

If your Splunk lab instance is Windows:

a. Use an RDC (Remote Desktop client) connection window to connect to your indexer using the designated host's domain name value **{DS-dns}**.



b. Open a remote desktop connection to the window and login using **{os-user}** (normally set to **student**, on Windows).



c. In the remote Windows desktop, click **Start > Command Prompt**.

d. When prompted with "The certificate couldn't be verified back to a root certificate. Your connection may not be secure. Do you want to continue?" click **Continue**.

Task 2: Retrieve Splunk settings from your deployment server using the CLI.

2. Navigate to the **SPLUNK_HOME** directory:

```
cd /opt/splunk/bin
```

```
cd "C:\Program Files\Splunk\bin"
```

3. Using the Splunk command line (CLI), run a command to check the status of your Splunk services.

```
./splunk status
```

```
splunk status
```

The output shows the running status and the **splunkd** process IDs:

```
splunkd is running (PID: #####)
splunk helpers are running (PIDs: #####,#####,…)
```

```
Splunkd: Running (pid #####)
```

4. Using the Splunk CLI, retrieve the following information about your Splunk server.

   If you are on the Windows server, omit the **./** from the commands. (For example, type: **splunk version**, instead of **./splunk version**)

   Use **splunk help commands** and **splunk help show** to view commands available and syntax help.

NOTE: You will be prompted for the Splunk administrator username (**admin**) and the password you were provided: **{password}**.

Splunk version	**./splunk version**
Splunk Web port:	**./splunk show web-port**
Splunk management (**splunkd**) port:	**./splunk show splunkd-port**
Splunk App Server ports:	**./splunk show appserver-ports**
Splunk KV store port:	**./splunk show kvstore-port**
Splunk server name:	**./splunk show servername**
Default host name:	**./splunk show default-hostname**

```
./splunk version
Splunk 8.2.0 (build ############)

./splunk show web-port
Your session is invalid.  Please login.
Splunk username: admin
Password: ********                                  (using the admin password {password})
Web port: 8000

./splunk show splunkd-port
Splunkd port: 8089

./splunk show appserver-ports
Application server ports on loopback interface: 8065

./splunk show kvstore-port
KV Store port: 8191

./splunk show servername
Server name: splunk##                               (where ## is your {student-ID})

./splunk show default-hostname
Default hostname for data inputs: splunk##.         (where ## is your {student-ID})
```

NOTE: When running commands in Splunk 9.x, you will see the following warning message:

```
WARNING: Server Certificate Hostname Validation is disabled. Please
see server.conf/[sslConfig]/cliVerifyServerName for details.
```

These messages can be safely ignored in your lab environment. See the section at the end of this lab titled Notes About Security and Splunk 9.x for more details.

Task 3: Examine Splunk configuration file documentation and basic `.conf` files

5. From your terminal window, navigate to the **`SPLUNK_HOME/etc/system`** directory:

```
cd /opt/splunk/etc/system
```

```
cd "C:\Program Files\Splunk\etc\system"
```

6. View the files in the **`README`** directory:

```
ls README
```

```
dir README
```

You should see a long list of Splunk configuration files, where each filename consists of a **`<filename>.conf.example`** file and a **`<filename>.conf.spec`** (specification) file.

7. View the files in the **`default`** directory:

```
ls default
```

```
dir default
```

You should see a long list of similarly named Splunk configuration (**`<filename>.conf`**) files. These are the default configuration files that ship with Splunk, and should not be modified.

8. View the files in the **`local`** directory:

```
ls local
```

```
dir local
```

You should see a much shorter list of Splunk configuration (**`<filename>.conf`**) files, such as **`inputs.conf`**, **`authentication.conf`**, **`server.conf`**, and **`web.conf`**. These are the configuration files that have been modified through initial Splunk installation and configuration, and possibly some of the steps you took in Splunk Web or using the command line in prior labs. These files can be modified.

9. View the **server.conf.spec** file in the **README** directory and review the documentation for the **[general]** stanza's **serverName** field.

```
more +39 README/server.conf.spec
```

```
more +39 README\server.conf.spec
```

> **NOTE:** Using the command **more +39** shows the file starting with line 39, which is where the documentation for the **General Server Configuration** section resides.
>
> Press the **Q** key to quit **more** output.

You should see the stanza and documentation starting with the following lines:

```
########################################################################
# General Server Configuration
########################################################################
[general]
serverName = <ASCII string>
* The name that identifies this Splunk software instance for features such as
  distributed search.
* Cannot be an empty string.
* Can contain environment variables.
...
```

10. View the **server.conf** file in the **default** directory and identify the value of the **serverName** field.

```
more +16 default/server.conf
```

The default value of the field on Linux should appear as:

```
serverName=$HOSTNAME
```

```
more +16 default\server.conf
```

The default value of the field on Windows should appear as:

```
serverName=$COMPUTERNAME
```

11. View the **server.conf** file in the **local** directory and identify the value of the **serverName** field.

```
more local/server.conf
```

```
more local\server.conf
```

The value of the field should match the server name you entered in Splunk Web under **Settings > Server Settings > General Settings** in a prior lab (where ## matches your student number):

```
serverName = splunk##
```

Task 4: Use Splunk commands to determine where the running configuration is coming from.

12. Navigate to the Splunk **bin** directory.

```
cd /opt/splunk/bin
```

```
cd "C:\Program Files\Splunk\bin"
```

13. Use the **splunk btool** command to view the **[general]** stanza settings for **server.conf** as they are found on the disk of the running Splunk server:

```
./splunk btool server list general
```

```
splunk btool server list general
```

14. Use the **splunk btool --debug** command to show which files these settings are coming from, taking note of the **serverName** setting.

```
./splunk btool server list general --debug
```

```
splunk btool server list general --debug
```

Examining the output, you can see that most parameters are set by **server.conf** in the **default** directory, however the **serverName** setting is set in the **local** directory:

```
...
/opt/splunk/etc/system/default/server.conf python.version = python3
/opt/splunk/etc/system/default/server.conf regex_cache_hiwater = 2500
/opt/splunk/etc/system/local/server.conf   serverName = splunk##
/opt/splunk/etc/system/default/server.conf sessionTimeout = 1h
...
```

15. Use the **splunk show config** command to view the **server.conf** configuration settings in memory on the running Splunk server:

```
./splunk show config server
```

```
splunk show config server
```

16. Use the **splunk show config server** command to examine just the **serverName** setting in memory:

```
./splunk show config server | grep serverName
```

```
splunk show config server | findstr serverName
```

The output shows the Splunk server name (where ## matches your student number):

```
serverName=splunk##
```

Task 5: View the input stanza created in Lab exercise 1 manually and using btool.

17. From your deployment server's command line (or text editor), review the contents of the **inputs.conf** file created by the **Add Data** wizard in Lab exercise 1, Task 6, and verify the following stanza:

```
more /opt/splunk/etc/apps/search/local/inputs.conf
```

```
[monitor:///opt/log/www2/access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie
```

```
more "C:\Program Files\Splunk\etc\apps\search\local\inputs.conf"
```

```
[monitor://C:\opt\log\www2\access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie
```

NOTE: If **Continuously Monitor** was not selected during the creation of your input, then Splunk does not create the above stanza.

18. Use the **btool** command to show all of the Splunk settings associated with the creation of the data input. In Windows, the entry for the drive letter (drive C in this task) is case sensitive.

NOTE: Remember to navigate to the **SPLUNK_HOME/bin** directory to run the **splunk** command.

```
./splunk btool inputs list monitor:///opt/log/www2/access.log
```

```
[monitor:///opt/log/www2/access.log]
_rcvbuf = 1572864
disabled = false
host = splunk##
index = test
sourcetype = access_combined_wcookie
```

```
splunk btool inputs list monitor://C:\opt\log\www2\access.log
```

```
[monitor://C:\opt\log\www2\access.log]
_rcvbuf = 1572864
disabled = false
evt_dc_name =
evt_dns_name =
evt_resolve_ad_obj = 0
host = splunk##
index = test
sourcetype = access_combined_wcookie
```

Notice that some attributes are shown using the **btool** command that do not appear in the **inputs.conf** file that we previously viewed, such as **host** and **_rcvbuf**.

19. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the data input.

```
./splunk btool inputs list monitor:///opt/log/www2/access.log --debug
```

```
/opt/splunk/etc/apps/search/local/inputs.conf [monitor:///opt/log/www2/access.log]
/opt/splunk/etc/system/default/inputs.conf    _rcvbuf = 1572864
/opt/splunk/etc/apps/search/local/inputs.conf disabled = false
/opt/splunk/etc/system/local/inputs.conf      host = splunk01
/opt/splunk/etc/apps/search/local/inputs.conf index = test
/opt/splunk/etc/apps/search/local/inputs.conf sourcetype = access_combined_wcookie
```

```
splunk btool inputs list monitor://C:\opt\log\www2\access.log --debug
```

```
C:\Program Files\Splunk\etc\apps\search\local\inputs.conf
     [monitor://C:\opt\log\www2\access.log]
C:\Program Files\Splunk\etc\system\default\inputs.conf    _rcvbuf = 1572864
C:\Program Files\Splunk\etc\apps\search\local\inputs.conf disabled = false
C:\Program Files\Splunk\etc\system\default\inputs.conf    evt_dc_name =
C:\Program Files\Splunk\etc\system\default\inputs.conf    evt_dns_name =
C:\Program Files\Splunk\etc\system\default\inputs.conf    evt_resolve_ad_obj = 0
C:\Program Files\Splunk\etc\system\local\inputs.conf      host = splunk01
C:\Program Files\Splunk\etc\apps\search\local\inputs.conf index = test
C:\Program Files\Splunk\etc\apps\search\local\inputs.conf sourcetype =
     access_combined_wcookie
```

NOTE: The **host** field shows the default hostname as defined in **SPLUNK_HOME/etc/system/local/inputs.conf** (on Linux) or **SPLUNK_HOME\etc\system\local\inputs.conf** (on Windows).

The **_rcvbuf** field shows the receive buffer default used for UDP port input. This field (as well as a few other fields on Windows) are defined in **SPLUNK_HOME/etc/system/default/inputs.conf** (on Linux) or **SPLUNK_HOME\etc\system\default\inputs.conf** (on Windows).

Task 6: Explore Splunk apps on Splunkbase.

In this task, you explore the Splunkbase website and view some of the Splunk apps currently available on that site.

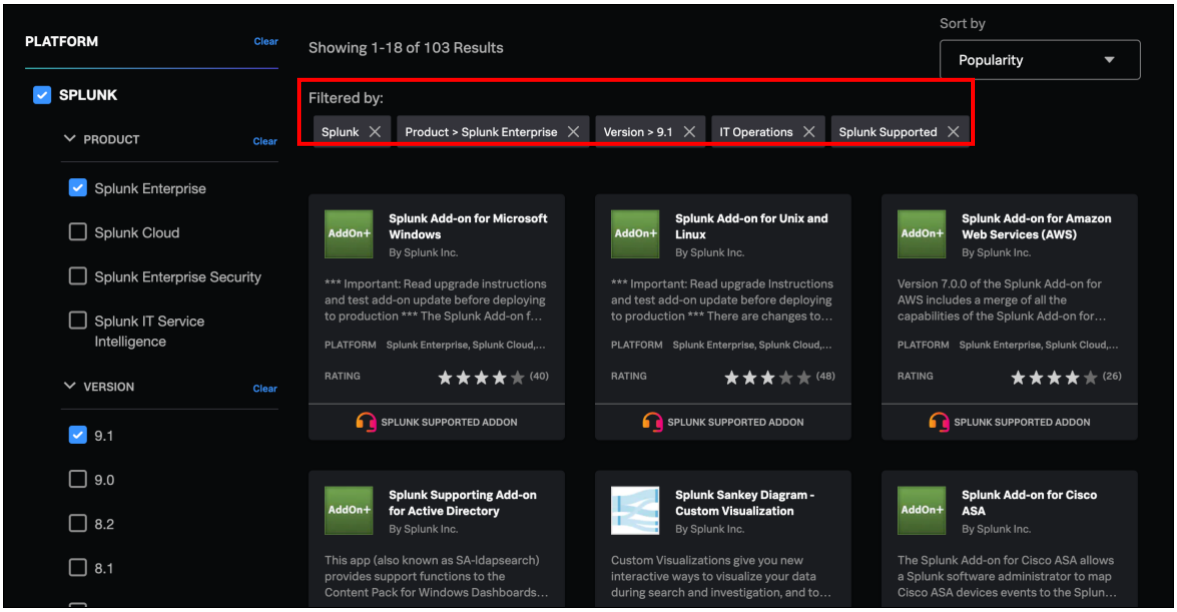
20. Visit **https://splunkbase.splunk.com/**.

NOTE: Note that to download any apps from Splunkbase, you first need a Splunk.com account. You do not need to create a Splunk.com account for this exercise.

21. At the top left of the page click **Apps**.

NOTE: This page can also be found directly at **https://splunkbase.splunk.com/apps**.

22. Search for apps that meet the following criteria by clicking the checkboxes on the left column of the page:
    - **PLATFORM > SPLUNK > PRODUCT >** Splunk Enterprise
    - **PLATFORM > SPLUNK > VERSION >** 9.1
    - **CATEGORIES:** IT Operations
    - **SUPPORT:** Splunk Supported



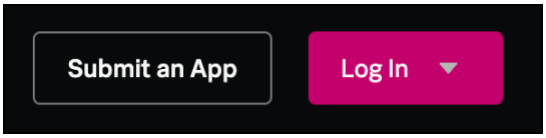
How many apps meet the above criteria? As of this writing, 103.

23. Click on the **Splunk Add-on for Unix and Linux**.

    This is one of the more popular add-ons and should appear towards the top of the page.

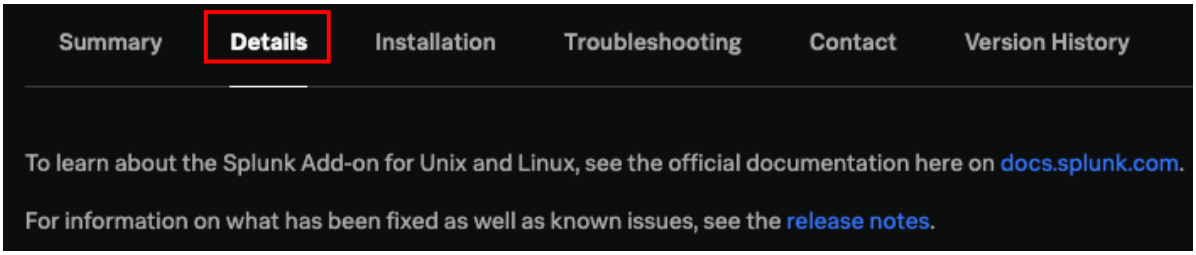
NOTE: This page can also be found directly at https://splunkbase.splunk.com/app/833.

24. If you are not logged into Splunkbase, notice the **Login to Download** button towards the top of the page.



You do not need to login or download the app at this time.

25. Scroll down to the tabs found lower in the page and click on **Details**.



Notice the links provided to the official documentation and release notes.

26. Optionally explore other areas and applications on the Splunkbase site, at your leisure.

Task 7: Explore currently installed apps by viewing configuration files.

27. From your terminal window, navigate to the **`SPLUNK_HOME/etc/apps`** directory:

```
cd /opt/splunk/etc/apps
```

```
cd "C:\Program Files\Splunk\etc\apps"
```

28. View the existing app directories for this Splunk server in **`SPLUNK_HOME/etc/apps`**:

```
ls
```

```
dir
```

You should see a number of directories including default apps such as **`launcher`** (the default Splunk Web **home** app), **`splunk_monitoring_console`** (the Monitoring Console) and **`search`** (the **Search & Reporting** app).

29. Display the app properties in the **`app.conf`** configuration file that determine the display name used for the **`search`** app and if the app is visible in Splunk Web.

```
more search/default/app.conf
```

```
more search\default\app.conf
```

These display properties are shown under the **`[ui]`** stanza:

```
...
[install]
is_configured = true
state = enabled
allows_disable = false
```

```
[ui]
is_visible = true
label = Search & Reporting
supported_themes = light,dark

[launcher]
author=Splunk
description=The Search app is Splunk's default interface for searching and analyzing
IT data. It allows you to index data into Splunk, add knowledge, build reports, and
create alerts. The Search app can be used across many areas of IT including
application management, operations management, security, and compliance.
...
```

30. Display the properties in the **indexes.conf** configuration file that reflect the **test** index you created in the first lab exercise.

```
more search/local/indexes.conf
```

```
more search\local\indexes.conf
```

These display properties are shown under the **[test]** stanza:

```
[test]
coldPath = $SPLUNK_DB/test/colddb
enableDataIntegrityControl = 0
enableTsidxReduction = 0
homePath = $SPLUNK_DB/test/db
maxTotalDataSizeMB = 512000
thawedPath = $SPLUNK_DB/test/thaweddb
```

31. Display the properties in the **inputs.conf** configuration file that reflect the input you created in the first lab exercise.

```
more search/local/inputs.conf
```

```
more search\local\inputs.conf
```

These display properties are shown under the **[monitor:/{path_to_input}]** stanza. For example on Linux:

```
[monitor:///opt/log/www2/access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie
```

32. View the existing app directories for this Splunk server in **SPLUNK_HOME/etc/deployment-apps**:

```
ls /opt/splunk/etc/deployment-apps
```

```
dir "C:\Program Files\Splunk\etc\deployment-apps"
```

You should see a **README** file. This directory is currently otherwise empty, however you will use this location to deploy apps to Splunk deployment clients later in this course.

33. View the **README** file in the **deployment-apps** directory.

```
more /opt/splunk/etc/deployment-apps/README
```

```
more "C:\Program Files\Splunk\etc\deployment-apps\README"
```

The **README** file states:

```
This directory is the default repository location for deployable apps in a
deployment server configuration.

For details on configuring as a deployment server, see
$SPLUNK_HOME/etc/system/README/serverclass.conf.spec, serverclass.conf.example or
the Admin manual at http://docs.splunk.com/Documentation.
```

Notes About Security and Splunk 9.x

When running commands in Splunk 9.x, you may see the following warning message:

```
WARNING: Server Certificate Hostname Validation is disabled. Please see
server.conf/[sslConfig]/cliVerifyServerName for details.
```

This message is concerning Splunk not being able to validate hosts due to a lack of Transport Layer Security (TLS) certificates while connecting to remote Splunk instances. This issue is described in more detail under security advisory SVD-2022-0606.

Currently Splunk does not prevent the use of these commands, but simply provides a warning about the security concerns of running without TLS certificates. In a later Splunk release, commands to remote Splunk servers (for example using the command option **-uri https://<splunkserver>:8089**) may be prevented from functioning without TLS certificates configured.

In a production environment the use of security certificates prevents these warning messages from occurring.

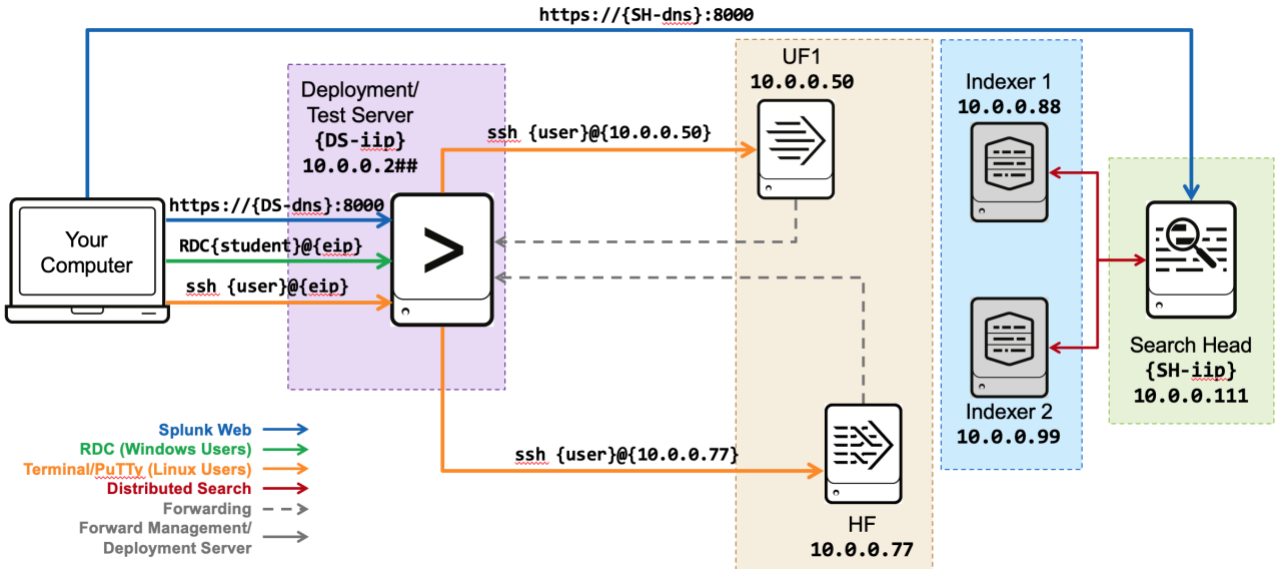
Splunk and certificate configuration for validation of Splunk commands is documented under the section "**Configure TLS host name validation for the Splunk CLI**" at: https://docs.splunk.com/Documentation/Splunk/latest/Security/EnableTLSCertHostnameValidation. This page also discusses enabling certification and hostname validation for other parts of the Splunk environment.

The topic of security is discussed in more detail in the Splunk Enterprise System Administration course.

Module 3 Lab – Configuring Forwarders

Description

In this exercise, you configure universal forwarder #1 (UF1, **10.0.0.50**) and your heavy forwarder (HF, **10.0.0.77**) to send data to your deployment/test server (**10.0.0.2##**) and validate the receipt of internal splunkd data on your deployment/test server.



Steps

Task 1: Configure receiving port on the Deployment/Test Server.

1. Connect to the command line of your dedicated Splunk deployment/test server.
2. Navigate to **SPLUNK_HOME**.

```
cd /opt/splunk/bin
```

```
cd "C:\Program Files\Splunk\bin"
```

3. View the current **splunktcp** settings in **inputs.conf** using the **btool** command.

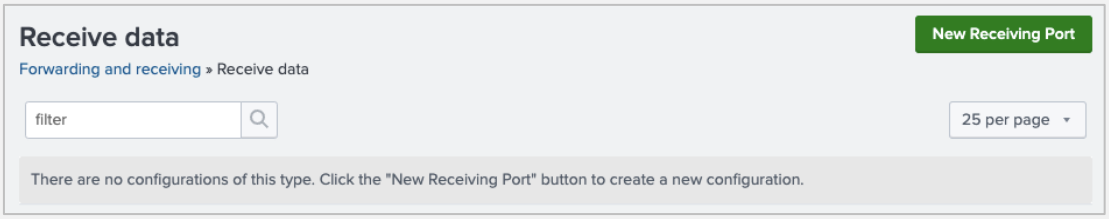
```
./splunk btool inputs list splunktcp --debug
```

```
splunk btool inputs list splunktcp --debug
```

You should see the following entries which configure some basic Splunk TCP settings, but reveal that no receiving port has been configured:

```
/opt/splunk/etc/system/default/inputs.conf    [splunktcp]
/opt/splunk/etc/system/default/inputs.conf    _rcvbuf = 1572864
/opt/splunk/etc/system/default/inputs.conf    acceptFrom = *
/opt/splunk/etc/system/default/inputs.conf    connection_host = ip
/opt/splunk/etc/system/local/inputs.conf      host = splunk01
/opt/splunk/etc/system/default/inputs.conf    index = default
/opt/splunk/etc/system/default/inputs.conf    logRetireOldS2S = true
/opt/splunk/etc/system/default/inputs.conf    logRetireOldS2SMaxCache = 10000
/opt/splunk/etc/system/default/inputs.conf    logRetireOldS2SRepeatFrequency = 1d
/opt/splunk/etc/system/default/inputs.conf    route = has_key:_replicationBucketUU
   ID:replicationQueue;has_key:_dstrx:typingQueue;has_key:_linebreaker:rulesetQueu
   e;absent_key:_linebreaker:parsingQueue
```

NOTE: You can also verify the receiving port configuration in Splunk Web under **Settings > Forwarding and Receiving > Configure receiving**. If no receiving ports are configured, you see the message "*There are no configuration of this type. Click the "New Receiving Port" button to create a new configuration.*" However in this lab you configure the receiving port using the command line.



4. Configure the receiving port on the deployment/test server with the **splunk enable listen** command.

```
./splunk enable listen 9997
```

```
splunk enable listen 9997
```

5. View the current **splunktcp** settings in **inputs.conf** using the **btool** command.

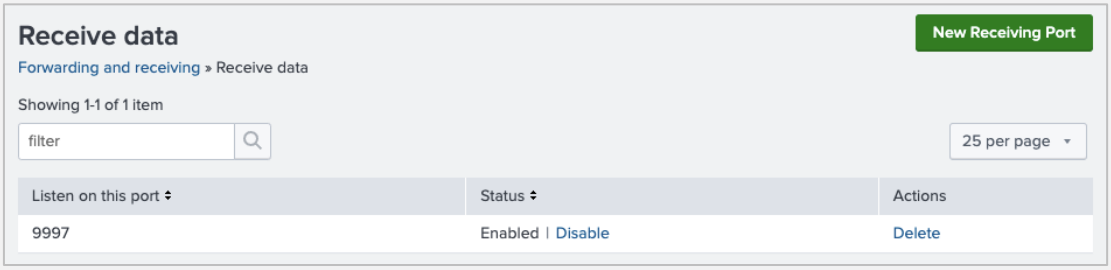
```
./splunk btool inputs list splunktcp --debug
```

```
splunk btool inputs list splunktcp --debug
```

In addition to the entries seen in the prior use of **btool**, you additionally see these entries specific to the **splunk enable listen** command you ran in the previous step:

```
...
/opt/splunk/etc/apps/search/local/inputs.conf [splunktcp://9997]
/opt/splunk/etc/system/default/inputs.conf    _rcvbuf = 1572864
/opt/splunk/etc/apps/search/local/inputs.conf connection_host = ip
/opt/splunk/etc/system/local/inputs.conf      host = splunk01
/opt/splunk/etc/system/default/inputs.conf    index = default
```

NOTE: You can also verify the receiving port configuration in Splunk Web under **Settings > Forwarding and Receiving > Configure receiving**. However in this lab you configure the receiving port using the command line.



6. View the **inputs.conf** settings found in the **search** app.

```
more /opt/splunk/etc/apps/search/local/inputs.conf
```

```
more "C:\Program Files\Splunk\etc\apps\search\local\inputs.conf"
```

The **inputs.conf** file shows the receiving port you just configured, in addition to the **access.log** input you configured on this local deployment/test server in a prior lab:

```
[monitor:///opt/log/www2/access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie

[splunktcp://9997]
connection_host = ip
```

Task 2: Connect to universal forwarder UF1.

7. From your deployment/test server, connect to the UF1 (**10.0.0.50**) using the following OS-specific instructions:

Use SSH to connect to your Linux depoyment/test server using IP address represented by **{DS-EIP}**. (You performed this in the prior lab.)

```
ssh {os-user}@{DS-EIP}
```

After establishing an SSH session to your deployment/test server, use SSH to connect to UF1 (**10.0.0.50**).

```
ssh {os-user}@10.0.0.50
```

When prompted for the authenticity of the host and the key fingerprint, type "yes" to continue.

Use an RDC (Remote Desktop client) connection window to connect to your Windows deployment/test server using the designated IP address value for **{DS-EIP}**. (You configured this in the prior lab.)

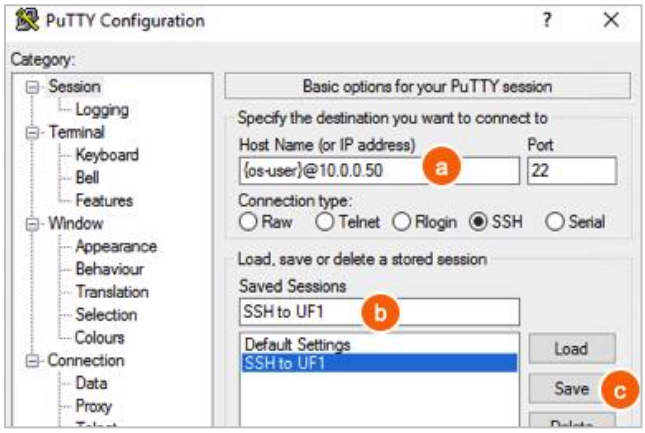
After connecting to your deployment/test server, locate **PuTTy** on the desktop:

Double-click the **PuTTy** application to open it, and configure an SSH session to UF1 with the following steps:

a. Replace **{os-user}@10.0.0.50** with your designated value of **{os-user}**.

b. Name your session, for example "**SSH to UF1**".

c. Save the session for later re-use.



d. Repeat steps a-c for other servers used in later lab steps:

- **{os-user}@10.0.0.100** with session name "**SSH to UF2**"
- **{os-user}@10.0.0.77** with session name "**SSH to HF**"

e. Click on the session "**SSH to UF1**" and click **Open** to start the session.

f. When prompted for the authenticity of the host and the key fingerprint, type "yes" to continue.

8. Verify that the command prompt indicates the location **ip-10-0-0-50**:

```
os-user@ip-10-0-0-50 ~]$
```

Task 3: Start and configure your universal forwarder instance, UF1.

9. To initialize the UF1, run the following commands:

```
cd ~/splunkforwarder/bin
./splunk start --accept-license
```

NOTE: This option automatically accepts the Splunk EULA. The **admin** password and the **splunkd-port** have already been configured for you. If you want to change your **splunkd-port**, you may need to check with your Splunk System Administrator and use **./splunk set splunkd-port <port_number>**.

10. After the installation, use the **show** command to view the **splunkd-port** number.

    Splunk will prompt you for the **admin** username and password.

```
./splunk show splunkd-port
Splunkd port: 1##89      (where ## is your student-ID)
```

11. Using the **set** command, change your forwarder's **servername** and the **default-hostname** to **engdev1##** where ## is your **{student-ID}**.

    This step uniquely identifies the data originating from your forwarder instance in this lab environment.

NOTE: Defer the restart until you have made all your changes.

```
./splunk set servername engdev1##              (where ## is your student-ID)
You need to restart the Splunk Server (splunkd) for your changes to take effect.

./splunk set default-hostname engdev1##        (where ## is your student-ID)
You need to restart the Splunk Server (splunkd) for your changes to take effect.
```

12. Restart UF1 to apply your changes.

```
./splunk restart
```

Task 4: Configure universal forwarder UF1 to send data directly to the deployment/test server.

In this task, you configure UF1 to send its internal Splunk logs directly to your deployment/test server.

13. Configure the forwarder to send data to port **9997** on a Splunk deployment/test server **10.0.0.2##**.

    Splunk will once again prompt you for the **admin** username and password, after the restart.

```
./splunk add forward-server 10.0.0.2##:9997
Added forwarding to: 10.0.0.2##:9997.
```

NOTE: You configured the listening port **9997** on your Splunk deployment/test server in Task 1.

14. Verify your forwarder is properly configured.

```
./splunk list forward-server
Active forwards:
        10.0.0.2##:9997
Configured but inactive forwards:
        None
```

NOTE: You may need to wait a few moments and run this command multiple times before you see the forward server is listed as Active.

15. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **outputs.conf** file.

```
./splunk btool outputs list tcpout:default-autolb-group --debug
```

```
/opt/home/{os_user}/splunkforwarder/etc/system/local/outputs.conf
   [tcpout:default-autolb-group]
/opt/home/{os_user}/splunkforwarder/etc/system/local/outputs.conf server =
   10.0.0.2##:9997
```

16. Restart UF1 to apply your new changes.

```
./splunk restart
```

17. Run **splunk list monitor** to show which file monitors are currently configured on UF1. You will be prompted to login as **admin**.

```
./splunk list monitor
```

    Currently only Splunk internal logs are being monitored:

```
Monitored Directories:
   $SPLUNK_HOME/var/log/splunk
      /opt/home/{user-ID}/splunkforwarder/var/log/splunk/audit.log
      /opt/home/{user-ID}/splunkforwarder/var/log/splunk/btool.log
      /opt/home/{user-ID}/splunkforwarder/var/log/splunk/conf.log
...
   $SPLUNK_HOME/var/run/splunk/search_telemetry/*search_telemetry.json
   $SPLUNK_HOME/var/spool/splunk/tracker.log*
Monitored Files:
   $SPLUNK_HOME/etc/splunk.version
```

18. Run **splunk add monitor** to add a new file monitor for **/opt/log/www2/access.log** on UF1.

```
./splunk add monitor /opt/log/www2/access.log -sourcetype access_combined_wcookie -index test
```

A confirmation message should appear:

```
Added monitor of '/opt/log/www2/access.log'.
```

19. Run **splunk list monitor** again to see any changes.

```
./splunk list monitor
```

The new monitored log file is listed at the bottom:

```
...
    $SPLUNK_HOME/var/run/splunk/search_telemetry/*search_telemetry.json
    $SPLUNK_HOME/var/spool/splunk/tracker.log*
Monitored Files:
    $SPLUNK_HOME/etc/splunk.version
    /opt/log/www2/access.log
```

20. View the resulting **inputs.conf** file and entries.

```
more ../etc/apps/search/local/inputs.conf
```

The monitor entry was created in the **search** app by default, and shows the index and source type:

```
[monitor:///opt/log/www2/access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie
```

21. Exit UF1's SSH session using the **exit** command.

Task 5: Configure the heavy forwarder (HF) to send data directly to the deployment/test server.

In this task, you configure HF to send its internal Splunk logs directly to your deployment/test server.

22. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**).

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session.

When prompted for the authenticity of the host and the key fingerprint, type "yes" to continue. Log in using your assigned password.

23. Verify that the command prompt indicates the location **ip-10-0-0-77**:

```
os-user@ip-10-0-0-77 ~]$
```

24. Navigate to the `bin` directory and initialize the forwarder with the **`--accept-license`** option.

```
cd ~/splunk/bin
./splunk start --accept-license
```

25. Use the CLI to determine the auto-assigned management port number.

    Splunk will prompt you for the **`admin`** username and password.

```
./splunk show splunkd-port
Splunkd port: 1##89
```

NOTE: This is the auto-assigned splunkd port. The ## is your student-ID number.

26. Using the **`set`** command, change your forwarder's **servername** and the **default-hostname** to **`splunkHF##`** where ## is your **`{student-ID}`**.

    This step uniquely identifies the data originating from your forwarder instance in this lab environment.

```
./splunk set servername splunkHF##
You need to restart the Splunk Server (splunkd) for your changes to take effect.

./splunk set default-hostname splunkHF##
You need to restart the Splunk Server (splunkd) for your changes to take effect.
```

NOTE: Defer the restarts until you have made all your changes.

27. Restart the forwarder.

```
./splunk restart
Stopping splunkd
...
Starting splunk server daemon (splunkd)...
Done
```

28. Configure the forwarder to send data to port **`9997`** on a Splunk deployment/test server **`10.0.0.2##`**.

    Splunk will once again prompt you for the **`admin`** username and password, after the restart.

```
./splunk add forward-server 10.0.0.2##:9997
Added forwarding to: 10.0.0.2##:9997.
```

NOTE: You configured the listening port **`9997`** on your Splunk deployment/test server in Task 1.

29. Verify your forwarder is properly configured.

```
./splunk list forward-server
Active forwards:
        10.0.0.2##:9997
Configured but inactive forwards:
        None
```

30. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **outputs.conf** file.

```
./splunk btool outputs list tcpout:default-autolb-group --debug
```

```
/opt/home/{os_user}/splunk/etc/system/local/outputs.conf [tcpout:default-autolb-
   group]
/opt/home/{os_user}/splunk/etc/system/local/outputs.conf server = 10.0.0.2##:9997
```

31. Restart HF to apply your new changes.

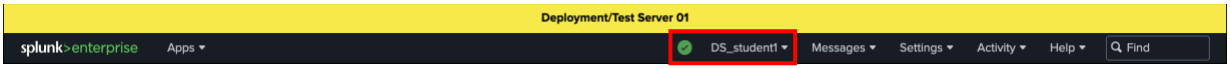
```
./splunk restart
```

32. Exit HF's SSH session using the **exit** command.

Task 6: Validate the receipt of forwarded data from your forwarders.

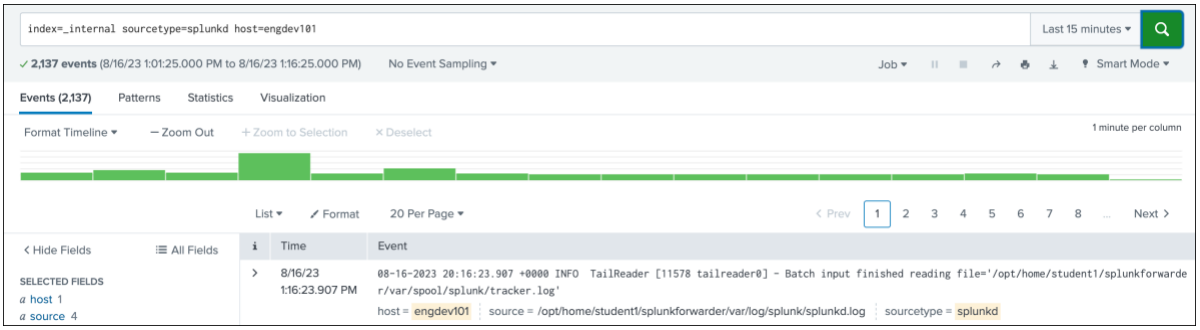
33. Log into Splunk Web for your deployment/test server.

    Remember that your deployment/test server is found at **https://{DS-dns}:8000**. To verify you are logged into the correct Splunk server, check that your username is listed as **DS_{user-ID}**.



34. Navigate to the **Search & Reporting** app.
35. Click **Skip/Skip tour** on any welcome windows that might display.
36. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 15 minutes**:

```
index=_internal sourcetype=splunkd host=engdev1##
```



    You should see events related to the **splunkd** process coming from **engdev1##**, your UF1.

37. Under the **INTERESTING FIELDS** column on the left, click on **splunk_server** and verify that it shows your deployment/test server.

38. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 15 minutes**:

    `index=_internal sourcetype=splunkd host=splunkHF##`

    You should see events related to the **splunkd** process coming from **`splunkHF##`**, your heavy forwarder.

39. Under the **INTERESTING FIELDS** column on the left, click on **splunk_server** and verify that it shows your deployment/test server.

40. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 24 hours**:

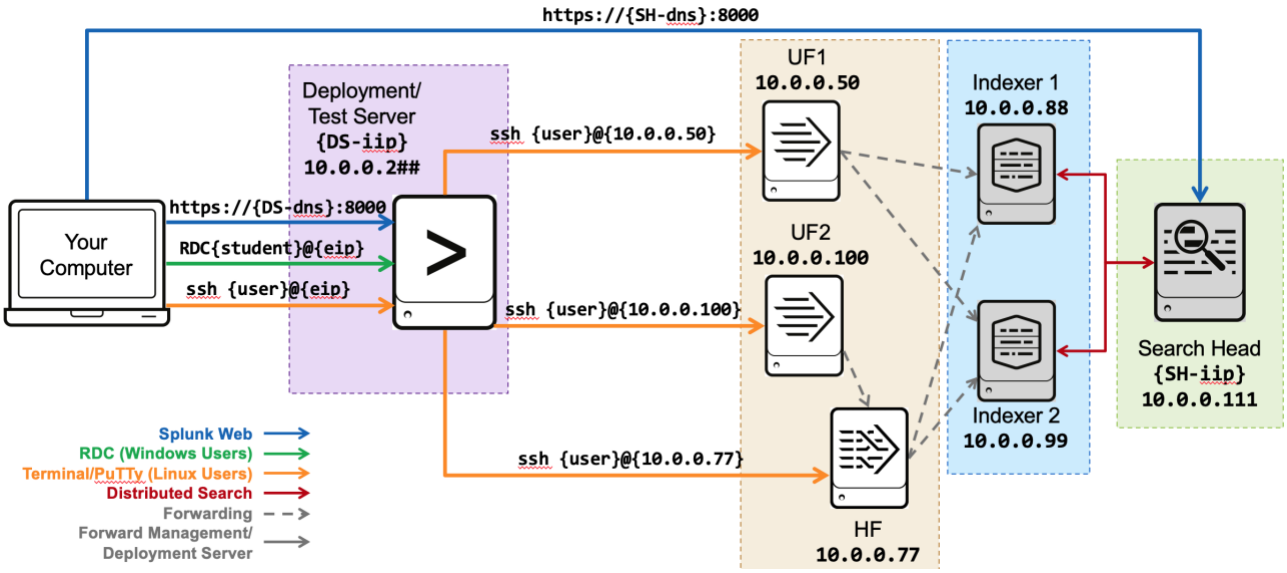
    `index=test host=engdev1##`

    You should see events related to the source **`/opt/log/www2/access.log`** coming from **`engdev1##`**, your universal forwarder UF1.

Module 4 Lab – Customizing Forwarders

Description

In this exercise, you change the configuration for universal forwarder #1 (UF1, **10.0.0.50**) and your heavy forwarder (HF, **10.0.0.77**) to send data to multiple, remote indexers (**10.0.0.88** and **10.0.0.99**) and validate the receipt of internal splunkd data on the shared search head. Additionally, you configure your heavy forwarder (HF, **10.0.0.77**) as an intermediate forwarder for universal forwarder #2 (UF2, **10.0.0.100**).



Steps

Task 1: Remove the deployment/test server as a forward-server for the universal forwarder, UF1

1. From your deployment/test server, connect to UF1 (**10.0.0.50**):

   After establishing an SSH session to your deployment/test server, use SSH to connect to UF1 (**10.0.0.50**). Log in using your assigned password.

   ```
   ssh {os-user}@10.0.0.50
   ```

   Open the **PuTTy** application, click on session "**SSH to UF1**" and click **Open** to start the session. Log in using your assigned password.

2. Navigate to the **bin** directory on the universal forwarder.

   ```
   cd ~/splunkforwarder/bin
   ```

3. List the current forward-server and verify it is currently set to your deployment/test server. If prompted, login as **admin**.

```
./splunk list forward-server
Active forwards:
        10.0.0.2##:9997
Configured but inactive forwards:
        None
```

4. Remove the deployment/test server as a forward-server.

```
./splunk remove forward-server 10.0.0.2##:9997
```

5. List the current forward-server and verify it is no longer set to your deployment/test server.

```
./splunk list forward-server
Active forwards:
        None
Configured but inactive forwards:
        None
```

Task 2: Configure your UF1 universal forwarder to send data directly to the two indexers.

In this task, you configure UF1 to send its internal Splunk logs, and any data it gathers in later lab exercises, directly to the pre-configured Splunk indexers.

6. Configure forwarder UF1 to send data to port **9997** on your Splunk indexers, **10.0.0.88** and **10.0.0.99**.

NOTE: The remote indexer ports on these indexers have been preconfigured to receive data.

```
./splunk add forward-server 10.0.0.88:9997
Added forwarding to: 10.0.0.88:9997.

./splunk add forward-server 10.0.0.99:9997
Added forwarding to: 10.0.0.99:9997.
```

7. Verify your forwarder is properly configured.

NOTE: The indexers will alternate between **Active** and **Configured but inactive forwards** due to load balancing. You may need to wait a minute and run the command multiple times to view these states.

```
./splunk list forward-server
Active forwards:
        None
Configured but inactive forwards:
        10.0.0.88:9997
        10.0.0.99:9997

./splunk list forward-server
Active forwards:
        10.0.0.88:9997
Configured but inactive forwards:
        10.0.0.99:9997
```

8. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **outputs.conf** file.

```
./splunk btool outputs list tcpout:default-autolb-group --debug
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf [tcpout:default-autolb-
group]
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf disabled = false
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf server =
10.0.0.88:9997,10.0.0.99:9997
```

9. Restart UF1 to apply your new changes.

```
./splunk restart
```

10. Exit UF1's SSH session.

```
exit
```

Task 3: Remove the deployment/test server as a forward-server for the heavy forwarder, HF

11. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

12. Navigate to the **bin** directory on the heavy forwarder.

```
cd ~/splunk/bin
```

13. List the current forward-server and verify it is currently set to your deployment/test server. If prompted, login as **admin**.

```
./splunk list forward-server
Active forwards:
        10.0.0.2##:9997
Configured but inactive forwards:
        None
```

14. Remove the deployment/test server as a forward-server.

```
./splunk remove forward-server 10.0.0.2##:9997
```

15. List the current forward-server and verify it is set to your deployment/test server.

```
./splunk list forward-server
Active forwards:
        None
Configured but inactive forwards:
        None
```

Task 4: Configure your HF heavy forwarder to send data directly to the two indexers.

In this task, you configure HF to send its internal Splunk logs, and any data it gathers in later lab exercises, directly to the pre-configured Splunk indexers.

16. Configure the forwarder to send data to port **9997** on your Splunk indexers, **10.0.0.88** and **10.0.0.99**.

NOTE: The remote indexer ports on these indexers have been preconfigured to receive data.

```
./splunk add forward-server 10.0.0.88:9997
Added forwarding to: 10.0.0.88:9997.

./splunk add forward-server 10.0.0.99:9997
Added forwarding to: 10.0.0.99:9997.
```

17. Verify your forwarder is properly configured.

NOTE: The indexers will alternate between **Active** and **Configured but inactive forwards** due to load balancing. You may need to wait a minute and run the command multiple times to view these states.

```
./splunk list forward-server
Active forwards:
        None
Configured but inactive forwards:
        10.0.0.88:9997
        10.0.0.99:9997

./splunk list forward-server
Active forwards:
        10.0.0.88:9997
Configured but inactive forwards:
        10.0.0.99:9997
```

18. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **outputs.conf** file.

```
./splunk btool outputs list tcpout:default-autolb-group --debug
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf [tcpout:default-autolb-
group]
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf disabled = false
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf server =
10.0.0.88:9997,10.0.0.99:9997
```

19. Restart HF to apply your new changes.

```
./splunk restart
```

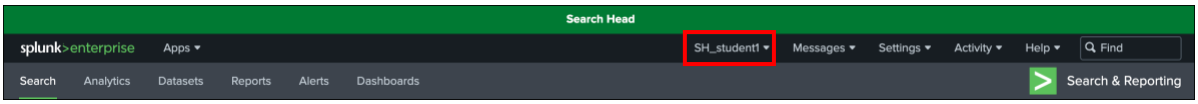
20. Exit HF's SSH session.

```
exit
```

Task 5: Validate the receipt of forwarded data from UF1 and HF.

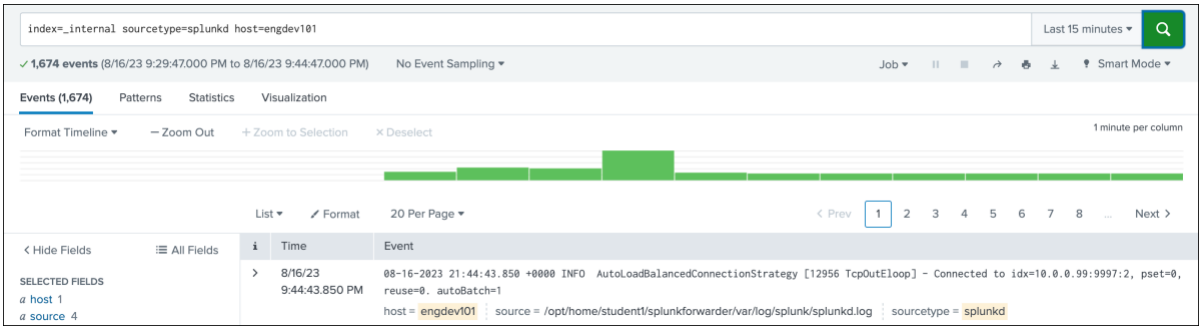
21. Log into the search head.

    Remember that your search head is found at **`https://{SH-dns}:8000`**. To verify you are logged into the search head, check that your username is listed as **`SH_{user-ID}`**.



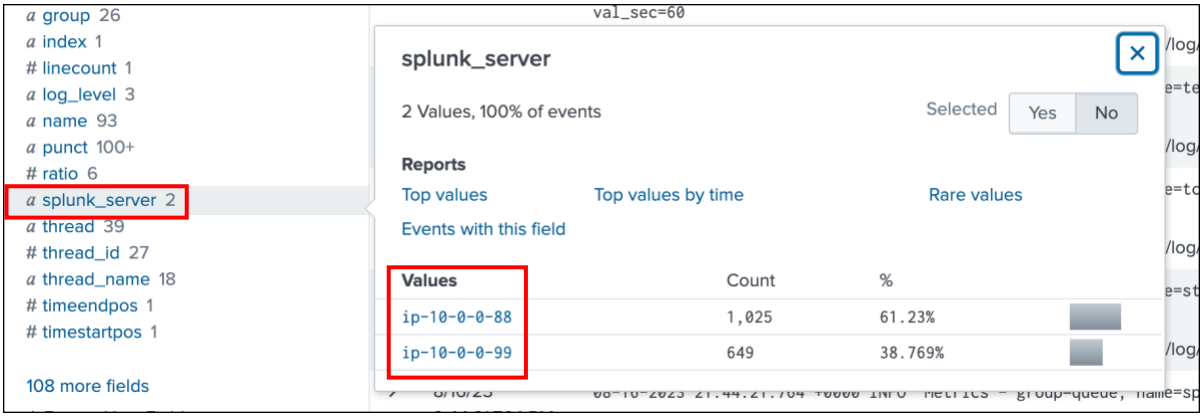
22. Navigate to the **Search & Reporting** app.
23. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 15 minutes**:

    `index=_internal sourcetype=splunkd host=engdev1##`



    You should see events related to the **splunkd** process coming from **`engdev1##`**, your UF1.

24. Under the **INTERESTING FIELDS** column on the left, click on **splunk_server** and verify that it shows the two indexers.



25. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 15 minutes**:

    `index=_internal sourcetype=splunkd host=splunkHF##`

    You should see events related to the **splunkd** process coming from **`splunkHF##`**, your heavy forwarder.

26. Under the **INTERESTING FIELDS** column on the left, click on **splunk_server** and verify that it shows the two indexers.

Task 6: Configure heavy forwarder HF to receive data as an intermediate forwarder.

In this task, you configure HF to receive data, so it can serve as an intermediate forwarder.

27. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

28. Navigate to the **bin** directory on the heavy forwarder.

```
cd ~/splunk/bin
```

29. Set up the receiving port on your HF to receive data from UF2, where ## is your **{student-id}**.

Splunk will prompt you for the **admin** username and password.

```
./splunk enable listen 99##
Listening for Splunk data on TCP port 99##.
```

30. View the configuration settings created by the **splunk enable listen** command.

```
more ~/splunk/etc/apps/search/local/inputs.conf
[splunktcp://9901]
connection_host = ip
```

31. Exit HF's SSH session.

```
exit
```

Task 7: Start and configure your universal forwarder instance, UF2.

In this task, you start universal forwarder UF2 and configure it to send its data to intermediate forwarder HF, which then sends the data to the two indexers.

32. From your deployment/test server, connect to UF2 (**10.0.0.100**):

After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**).

```
ssh {os-user}@10.0.0.100
```

Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session.

When prompted for the authenticity of the host and the key fingerprint, type "yes" to continue. Log in using your assigned password.

33. Verify that the command prompt indicates the location **ip-10-0-0-100**:

```
os-user@ip-10-0-0-100~]$
```

34. To initialize the universal forwarder UF2, run the following commands:

```
cd ~/splunkforwarder/bin
./splunk start --accept-license
```

NOTE: This option automatically accepts the Splunk EULA. The **admin** password and the **splunkd-port** have already been configured for you. If you want to change your **splunkd-port**, you may need to check with your Splunk System Administrator and use **./splunk set splunkd-port <port_number>**.

35. After the installation, use the **show** command to view the **splunkd-port** number.

    Splunk will prompt you for the **admin** username and password.

```
./splunk show splunkd-port
Splunkd port: 1##89      (where ## is your student-ID)
```

36. To uniquely identifies the data originating from your forwarder instance in this lab environment, use the **set servername** and **set default-hostname** commands, to change your forwarder's hostname to **engdev2##**, where ## is your **{student-id}**:

NOTE: Defer the restarts until you have made all your changes.

```
./splunk set servername engdev2##
You need to restart the Splunk Server (splunkd) for your changes to take effect.

./splunk set default-hostname engdev2##
You need to restart the Splunk Server (splunkd) for your changes to take effect.
```

37. Configure UF2 to send data the heavy forwarder HF at **10.0.0.77** on port **99##**.

    Note that Splunk will once again prompt you for the **admin** username and password, after the restart.

```
./splunk add forward-server 10.0.0.77:99##
Added forwarding to: 10.0.0.77:99##.
```

38. Verify your forwarder is properly configured.

```
./splunk list forward-server
Active forwards:
        10.0.0.77:99##
Configured but inactive forwards:
        None
```

39. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **outputs.conf** file.

```
./splunk btool outputs list tcpout:default-autolb-group --debug
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf [tcpout:default-autolb-
group]
/opt/home/os_user/splunkforwarder/etc/system/local/outputs.conf server = 10.0.0.77:99##
```

40. Restart UF2 to apply your new changes.

```
./splunk restart
```

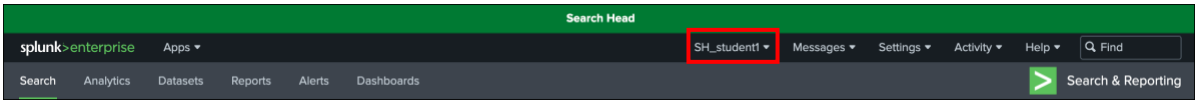
41. Exit UF2's SSH session.

```
exit
```

Task 8: Validate the receipt of forwarded data from UF2.

42. Log into the search head.

    Remember that your search head is found at **`https://{SH-dns}:8000`**. To verify you are logged into the search head, check that your username is listed as **`SH_{user-ID}`**.



43. Navigate to the **Search & Reporting** app.
44. Enter the search below. Replace the ##'s with your student ID and execute the following search over the **Last 24 hours**:

```
index=_internal sourcetype=splunkd host=engdev2## | stats count by host
```

    You should see events related to the **splunkd** process coming from **`engdev2##`**, your UF2.

45. Under the **count** column, click the event count, then click **View events** to switch from displaying Statistics to displaying the actual events.
46. Under the **INTERESTING FIELDS** column on the left, click on **splunk_server** and verify that it shows the two indexers.

NOTE: The internal Splunk log data from UF2 was sent to intermediate forwarder HF, which then forwarded the data to the two Splunk indexers (**`10.0.0.88`** and **`10.0.0.99`**). You have verified this by searching for the data using the Splunk search head, which is pulling that data from both indexers.

Module 5 Lab – Managing Forwarders

Description

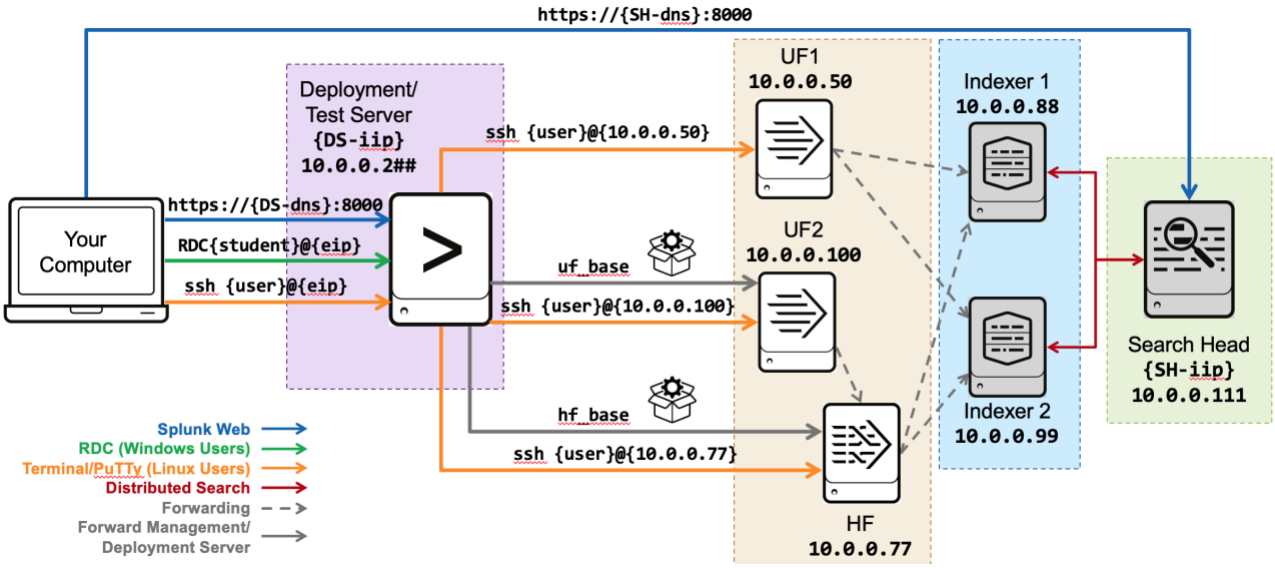
In this exercise, you will use the Forwarder Management interface in Splunk Web on the deployment server to configure a remote universal forwarder and a heavy forwarder. The advantage of this option is that it allows you to manage multiple groups of forwarders from a central location.

First, you will undo the forwarding of UF2 and HF that was configured in the prior lab. Then, you will enable the deployment server feature on your deployment server and stage two deployable apps for your forwarders (that have already been created for you) which will re-establish the forwarding configuration using the **`outputs.conf`** file to tell the fowarder where to send its data. The apps, **`uf_base`** for universal forwarder #2 (UF2) and **`hf_base`** for the heavy forwarder, are staged in **`SPLUNK_HOME/etc/deployment-apps`**.

Next, you will configure universal forwarder UF2 (**`10.0.0.100`**) and heavy forwarder HF (**`10.0.0.77`**) as deployment clients.

Finally, you will define a **serverclass** in the Forwarder Management UI of the deployment server to deploy the **`uf_base`** and **`hf_base`** apps to their correct forwarders. The serverclass associates deployable apps with deployment clients.

IMPORTANT:	Completing this lab exercise is crucial because it is a prerequisite to several subsequent lab exercises.



Steps

Task 1: Remove forwarding configuration from universal forwarder UF2

One goal of this lesson is to manage forwarding configuration from the deployment server for some of our deployment clients.

The forwarding configuration for universal forwarder UF2 and heavy forwarder HF is currently stored in the **outputs.conf** file of **SPLUNK_HOME/etc/system/local**. Due to Splunk configuration file precedence rules, these configuration settings cannot be overridden by other configuration files, which prevents us from managing these settings from the Deployment server. We must start by removing these forwarding configuration settings from **SPLUNK_HOME/etc/system/local/outputs.conf**.

1. From your deployment/test server, connect to UF2 (**10.0.0.100**):

   After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**). Log in using your assigned password.

   ```
   ssh {os-user}@10.0.0.100
   ```

   Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session. Log in using your assigned password.

2. Navigate to the **bin** directory on the universal forwarder.

   ```
   cd ~/splunkforwarder/bin
   ```

3. Verify UF2 is currently configured to forward data to heavy forwarder HF (**10.0.0.77**). You may need to log in as **admin**.

   ```
   ./splunk list forward-server
   Active forwards:
           10.0.0.77:99##
   Configured but inactive forwards:
           None
   ```

4. Use the **splunk btool** command to identify the location of the forward server configuration.

   ```
   ./splunk btool outputs list tcpout:default --debug
   /opt/home/{os_user}/splunkforwarder/etc/system/local/outputs.conf [tcpout:default-
   autolb-group]
   /opt/home/{os_user}/splunkforwarder/etc/system/local/outputs.conf server =
   10.0.0.77:99##
   ```

5. View the **outputs.conf** configuration file where these settings are stored.

   ```
   more ~/splunkforwarder/etc/system/local/outputs.conf
   [tcpout]
   defaultGroup = default-autolb-group

   [tcpout:default-autolb-group]
   server = 10.0.0.77:99##

   [tcpout-server://10.0.0.77:99##]
   ```

6. Remove the existing forward server configuration using the **splunk remove forward-server** command.

```
./splunk remove forward-server 10.0.0.77:99##
Stopped forwarding to: 10.0.0.77:99##.
```

7. View the **outputs.conf** configuration file after running this command.

```
more ~/splunkforwarder/etc/system/local/outputs.conf
[tcpout]
defaultGroup =
```

NOTE: In this case Splunk leaves behind an entry in **SPLUNK_HOME/etc/system/local** that prevents you from managing these configuration settings using a deployment app in the future. Best practice would be to remove this entry. Since this is the only entry in your **outputs.conf** file, it is simpler to remove the file.

8. Remove the **outputs.conf** configuration file.

```
rm ~/splunkforwarder/etc/system/local/outputs.conf
```

9. Exit the universal forwarder using the **exit** command.

Task 2: Remove forwarding configuration from heavy forwarder HF

10. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

11. Navigate to the **bin** directory on the heavy forwarder.

```
cd ~/splunk/bin
```

12. Verify HF is currently configured to forward data to the indexers (**10.0.0.88** and **10.0.0.99**). You may need to log in as **admin**.

```
./splunk list forward-server
Active forwards:
        10.0.0.88:9997
Configured but inactive forwards:
        10.0.0.99:9997
```

13. Use the **splunk btool** command to identify the location of the forward server configuration.

```
./splunk btool outputs list tcpout:default --debug
/opt/home/{os_user}/splunk/etc/system/local/outputs.conf [tcpout:default-autolb-group]
/opt/home/{os_user}/splunk/etc/system/local/outputs.conf server =
10.0.0.88:9997,10.0.0.99:9997
```

14. View the **outputs.conf** configuration file where these settings are stored.

```
more ~/splunk/etc/system/local/outputs.conf
[tcpout]
defaultGroup = default-autolb-group

[tcpout-server://10.0.0.88:9997]

[tcpout:default-autolb-group]
disabled = false
server = 10.0.0.88:9997,10.0.0.99:9997

[tcpout-server://10.0.0.99:9997]
```

15. Remove the existing forward server configuration using the **splunk remove forward-server** command for both indexers.

```
./splunk remove forward-server 10.0.0.88:9997
Stopped forwarding to: 10.0.0.88:9997.

./splunk remove forward-server 10.0.0.99:9997
Stopped forwarding to: 10.0.0.99:9997.
```

16. View the **outputs.conf** configuration file after running this command.

```
more ~/splunk/etc/system/local/outputs.conf
[tcpout]
defaultGroup =
```

17. Remove the **outputs.conf** configuration file.

```
rm ~/splunk/etc/system/local/outputs.conf
```

18. Exit the heavy forwarder using the **exit** command.

Task 3: Copy the uf_base app to the deployment-apps directory and configure outputs.conf.

In this first task, you will copy the **uf_base** app and stage the app to be deployed to UF2. The **outputs.conf** file will be configured to send its data to the receiving port of the heavy forwarder.

19. Access your deployment server's command line (SSH for Linux, RDC for Windows).

```
os-user@ip-10-0-0-2xx ~]$
```

20. Copy the entire **uf_base** directory from /opt/apps to **SPLUNK_HOME/etc/deployment-apps/**

```
cp -r /opt/apps/uf_base /opt/splunk/etc/deployment-apps/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
xcopy /S /I /E \opt\apps\uf_base "C:\Program Files\Splunk\etc\deployment-apps\uf_base"
```

21. View the **local** directory of the **uf_base** app (in **deployment-apps**) and list its contents to ensure the **outputs.conf** file was copied successfully.

```
ls /opt/splunk/etc/deployment-apps/uf_base/local
```

Use the Windows file browser to navigate to the new folder. Or, run:

```
dir "C:\Program Files\Splunk\etc\deployment-apps\uf_base\local"
```

NOTE: You will also see a **deploymentclient.conf** file in the local directory. This file is also deployed to the forwarder to reduce the polling interval (how often the deployment client contacts the deployment server) from 60 seconds (default) to 30 seconds:

```
[deployment-client]
phoneHomeIntervalInSecs = 30
```

22. Using the appropriate editor, open the newly copied **outputs.conf**:

```
/opt/splunk/etc/deployment-apps/uf_base/local/outputs.conf
```

```
"C:\Program Files\Splunk\etc\deployment-apps\uf_base\local\outputs.conf"
```

Linux users can use **vi** or **nano**, Windows users can use **Notepad++**

NOTE: **Windows Users**: Be aware that you must have administrator rights when editing the Splunk configuration files. In some environments you may need to launch the application you are using (for example **Notepad++**) with administrator rights. Additionally, be aware that you need to save the configuration files with the correct file extensions:

- Right-click the **Notepad++** and select **Run as administrator**.
- When saving files, click **Save as** and use the **All types (*.*)** option.
  Do not save your files as text files (*.txt files).

23. Add the stanza below to the **outputs.conf** file by replacing ## with your {**student-ID**}:

NOTE: Most Splunk configuration file contents are case-sensitive. If you copy and paste from the PDF lab document to the configuration files, ensure the contents are exactly as shown in the steps.

NOTE: Before adding the below stanzas to the file, the outputs.conf contains no entries.

```
[tcpout]
defaultGroup = default-autolb-group

[tcpout-server://10.0.0.77:99##]

[tcpout:default-autolb-group]
disabled = false
server = 10.0.0.77:99##
```

24. Save and close the edited file.
25. Verify the **outputs.conf** file was edited correctly by viewing its contents.

```
cat /opt/splunk/etc/deployment-apps/uf_base/local/outputs.conf
```

```
type "C:\Program Files\Splunk\etc\deployment-apps\uf_base\local\outputs.conf"
```

NOTE: If the file does not contain the contents from step 5, redo the steps in this lab task before continuing with the lab.

Task 4: Configure universal forwarder #2 (UF2) as a deployment client.

In this task, you manually configure a forwarder as a deployment client by using the **splunk set deploy-poll** command. (Another option is to include a **deploymentclient.conf** file with the proper settings into pre-configured software builds for universal forwarders.)

NOTE: Since many Splunk environments use hundreds or thousands of forwarders, these manual configurations may not be practical or scalable. Many Splunk customers use a third-party software configuration management tool, such as Puppet or Chef.

26. From your deployment/test server, connect to UF2 (**10.0.0.100**):

After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.100
```

Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session. Log in using your assigned password.

27. Navigate to the **bin** directory.

```
cd ~/splunkforwarder/bin
```

28. Use the set **deploy-poll** command to establish communication between the forwarder and the deployment server, where ## is your **{student-id}**. You may need to log in as **admin**.

```
./splunk set deploy-poll 10.0.0.2##:8089
Configuration updated.
```

29. Restart the forwarder to have it quickly check in with the DS in the Forwarder Management view.

```
./splunk restart
Stopping splunkd
...

Starting splunk server daemon (splunkd)...
Done
```

30. Use the **show deploy-poll** command to verify the deployment-client configuration.

    Splunk will prompt you for the **admin** username and password.

```
./splunk show deploy-poll
Deployment Server URI is set to "10.0.0.2##:8089"
```

NOTE: **10.0.0.2##** is the internal address of your deployment server instance.

31. Use the **btool** command with the **--debug** flag to show all of the Splunk settings associated with the creation of the **deploymentclient.conf** file.

```
./splunk btool deploymentclient list --debug
/opt/home/{os_user}/splunkforwarder/etc/system/local/deploymentclient.conf
[target-broker:deploymentServer]
/opt/home/{os_user}/splunkforwarder/etc/system/local/deploymentclient.conf
targetUri = 10.0.0.2##:8089
```

32. Exit UF2's SSH session.

```
exit
```

Task 5: Copy the hf_base app to the deployment-apps directory and configure outputs.conf.

Copy the **hf_base** app and stage the app to be deployed to heavy forwarder. The **outputs.conf** file has been pre-configured so that the heavy forwarder will send its data to the receiving ports of the remote indexers.

33. Access your deployment server's command line (SSH for Linux, RDC for Windows).

```
os-user@ip-10-0-0-2xx ~]$
```

34. Copy the entire **hf_base** directory from /opt/apps to **SPLUNK_HOME/etc/deployment-apps/**

```
cp -r /opt/apps/hf_base /opt/splunk/etc/deployment-apps/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
xcopy /S /I /E \opt\apps\hf_base "C:\Program Files\Splunk\etc\deployment-apps\hf_base"
```

35. Navigate to the **local** directory of the **hf_base** app in **deployment-apps** and list its contents to make sure the **outputs.conf** file was copied successfully.

```
ls /opt/splunk/etc/deployment-apps/hf_base/local
```

Use the Windows file browser to navigate to the new folder. Or, run:

```
dir "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

36. View the contents of the **outputs.conf** file:

```
more /opt/splunk/etc/deployment-apps/hf_base/local/outputs.conf
```

Either run:

```
dir "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

Or use the Windows file browser view the file contents:

- Right-click the **Notepad++** and select **Run as administrator**.
- Select **File > Open** and navigate to **C:\Program Files\Splunk\etc\deployment-apps\hf_base\local** and open the **outputs.conf** file.

```
[tcpout]
defaultGroup = default-autolb-group

[tcpout-server://10.0.0.88:9997]

[tcpout-server://10.0.0.99:9997]

[tcpout:default-autolb-group]
disabled = false
server = 10.0.0.88:9997,10.0.0.99:9997
```

Task 6: Configure the heavy forwarder (HF) as a deployment client.

Enable the listening port on the heavy forwarder (HF) to listen for Splunk data being transmitted from UF2. Then, manually configure the HF as a deployment client by using the **`splunk set deploy-poll`** command.

37. From your deployment/test server, connect to HF (**`10.0.0.77`**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

38. Navigate to the `bin` directory on the HF.

```
cd ~/splunk/bin
```

39. Use the **`set deploy-poll`** command to establish communication between the forwarder and the deployment server, where ## is your **`{student-id}`**. You may need to log in as **admin**.

```
./splunk set deploy-poll 10.0.0.2##:8089
Configuration updated.
```

40. Restart the forwarder to have it quickly check in with the DS in the Forwarder Management view.

```
./splunk restart
Stopping splunkd
...

Starting splunk server daemon (splunkd)...
Done
```

41. Use the **`show deploy-poll`** command to verify the deployment-client configuration.

    Splunk will prompt you for the **`admin`** username and password.

```
./splunk show deploy-poll
Deployment Server URI is set to "10.0.0.2##:8089"
```

NOTE: **`10.0.0.2##`** is the internal address of your deployment server instance.

42. Use the **btool** command with the **`--debug`** argument to show all of the Splunk settings associated with the creation of the **`deploymentclient.conf`** file.

```
./splunk btool deploymentclient list --debug
/opt/home/{os-user}/splunk/etc/system/local/deploymentclient.conf [target-
broker:deploymentServer]
/opt/home/{os-user}/splunk/etc/system/local/deploymentclient.conf targetUri =
10.0.0.2##:8089
```

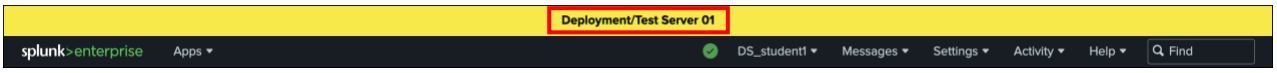
43. Exit HF's SSH session using the **`exit`** command.

```
exit
```

Task 7: Add a server class to manage the HF from your deployment server.

You should now have two deployment apps to deploy and two deployment clients running and waiting to receive deployment apps. To complete the forwarder management enablement, you will need to configure a server class. The server class will associated the apps with the appropriate deployment client. In this task, you will create a server class for the HF client and assign it the **`hf_base`** app.

44. Log into Splunk Web as **`admin`** on the deployment server.



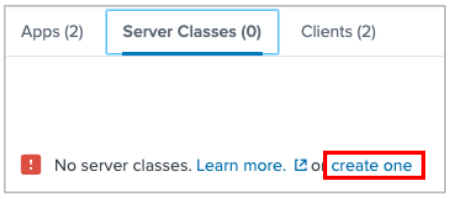
45. Navigate to **Settings > Forwarder management**.

    If **Forwarder Management** is not found in the menu, verify you are on the deployment server.

46. Select the **Apps** tab. The **`hf_base`** and **`uf_base`** apps should display.
47. Select the **Clients** tab. Hosts **`ip-10-0-0-77`** (heavy forwarder) and **`ip-10-0-0-100`** (UF2) should display.

NOTE: It can take several minutes before your clients appears in the user interface. Proceed to the next steps while waiting for the full connection.

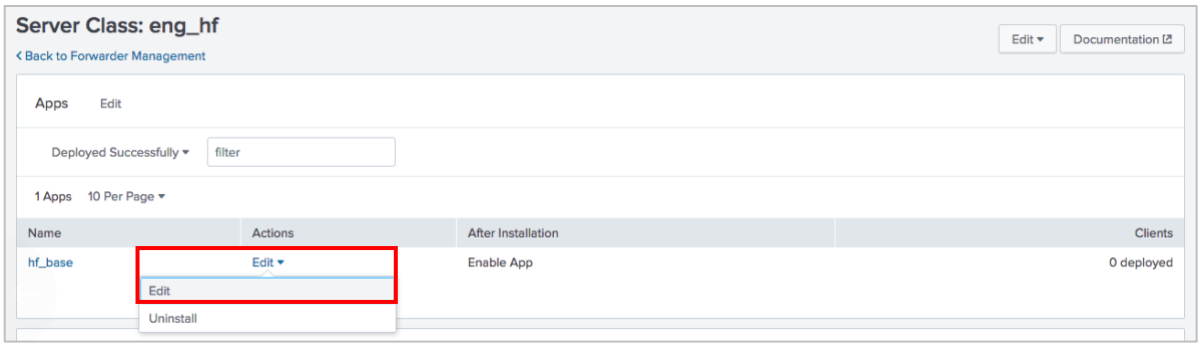
48. On the **Server Classes** tab, create a new Server Class by clicking on **create one**.



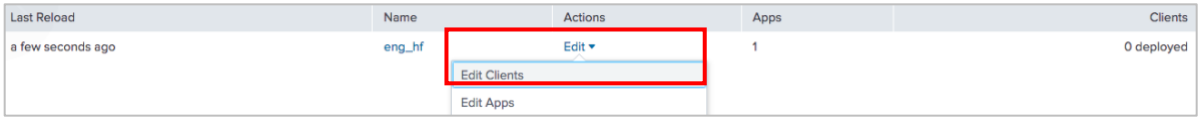
49. In the **New Server Class** window, name the server class **`eng_hf`** and click **Save**.
50. Click **Add Apps**.
51. Click the **`hf_base`** app to move it to the **Selected Apps** panel, then click **Save**.

    In the **After Installation** column of the **hf_base** app, it shows **Enable App**.

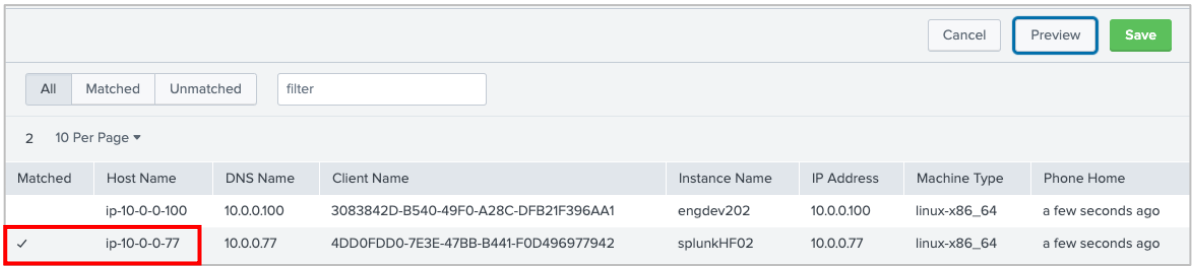
52. Under the **Actions** column click **Edit** > **Edit.**



53. On the **Edit App: hf_base** page, select the **Restart Splunkd** check box, then click **Save**.
54. Select the **Server Classes** tab.
55. Under the **Actions** column, click **Edit > Edit Clients**.



56. Enter the deployment client's IP address **`10.0.0.77`** to the **Include (Includelist)** box.
57. Click **Preview**.
58. When the check mark appears in the **Matched** column for host **`ip-10-0-0-77`**, click **Save**.



Task 8: Add a server class to manage UF2 from your deployment server.

Create a server class for UF2 and assign the **`uf_base`** app.

59. From the **Server Classes** tab, click **New Server Class**.
60. In the **New Server Class** window, name the server class **`eng_uf`** and click **Save**.
61. Click **Add Apps**.
62. Click the **`uf_base`** app to move it to the **Selected Apps** panel, then click **Save**.

    In the **After Installation** column of the **uf_base** app, it shows **Enable App**.

63. Under the **Actions** column click **Edit** > **Edit.**
64. On the **Edit App: uf_base** page, select the **Restart Splunkd** check box, then click **Save**.
65. Click the **Server Classes** tab.
66. Under the **Actions** column, click **Edit > Edit Clients** of the **eng_uf** server class.
67. Enter the deployment client's IP address **`10.0.0.100`** to the **Include (Includelist)** box.
68. Click **Preview**.
69. When the check mark appears in the **Matched** column for host **`ip-10-0-0-100`**, click **Save**.

Check Your Work

Task 9: Confirm the deployment of the hf_base app on the Heavy Forwarder HF.

70. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

71. From your HF terminal window, confirm that the directory **hf_base** exists in **~/splunk/etc/apps**.

```
cd ~/splunk/etc/apps
ls -t
hf_base                         introspection_generator_addon
splunk_monitoring_console       journald_input
...
```

72. Verify that the **outputs.conf** file matches the following:

```
cat ~/splunk/etc/apps/hf_base/local/outputs.conf

[tcpout]
defaultGroup = default-autolb-group

[tcpout-server://10.0.0.99:9997]

[tcpout-server://10.0.0.88:9997]

[tcpout:default-autolb-group]
disabled = false
server = 10.0.0.99:9997,10.0.0.88:9997
```

73. Verify HF is currently configured to forward data to the indexers (**10.0.0.88** and **10.0.0.99**). You may need to log in as **admin**.

```
cd ~/splunk/bin
./splunk list forward-server
Active forwards:
        10.0.0.88:9997
Configured but inactive forwards:
        10.0.0.99:9997
```

74. Exit the SSH session

```
exit
```

Task 10: Confirm the deployment of the uf_base app on the Universal Forwarder UF2.

75. From your deployment/test server, connect to UF2 (**10.0.0.100**):

After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.100
```

Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session. Log in using your assigned password.

76. From your UF2 terminal window, confirm directory **uf_base** exists in **~/splunkforwarder/etc/apps**.

```
cd ~/splunkforwarder/etc/apps
ls -t
uf_base                        journald_input    splunk_internal_metrics
learned                        search            SplunkUniversalForwarder
introspection_generator_addon  splunk_httpinput
```

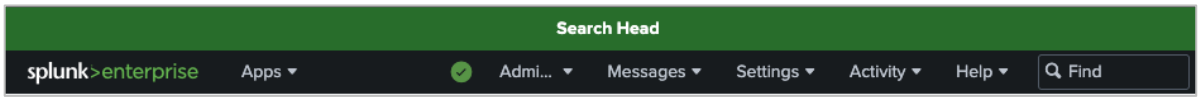
77. Verify that the **outputs.conf** file matches the following:

```
cat ~/splunkforwarder/etc/apps/uf_base/local/outputs.conf

[tcpout]
defaultGroup = default-autolb-group

[tcpout-server://10.0.0.77:99##]

[tcpout:default-autolb-group]
disabled = false
server = 10.0.0.77:99##
```

78. Exit the SSH session

```
exit
```

79. Navigate to the search head: **https://{SH-dns}:8000**.
80. Log in with your assigned **{user-ID}** and password **{password}**.
81. Verify that a green banner appears above the Splunk Web navigation bar.

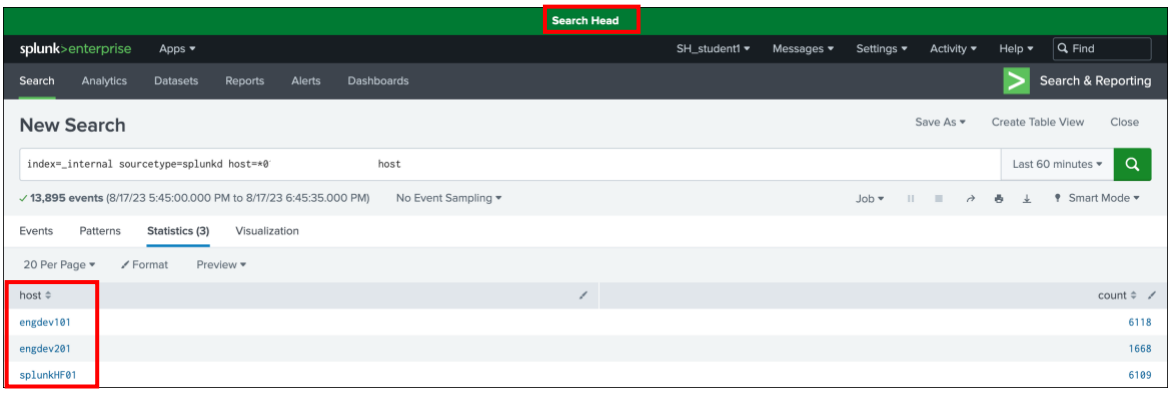


82. In Splunk Web on the search head execute the following search over the **Last 15 minutes** (replace the ## with your student ID):

```
index=_internal sourcetype=splunkd host=*## | stats count by host
```

83. You should now see following hosts (where ## = student id):

- **engdev1##** (UF1)
- **engdev2##** (UF2)
- **splunkHF##** (Heavy Forwarder)



NOTE: If you do not see the correct hosts, ensure you are running the search in Splunk Web on the search head (and not on the deployment server).

Troubleshooting Suggestions

If your deployment is not indexing the internal events from UF2 and the heavy forwarder, check the following:

1. A common error is running the forwarder commands on the deployment server. In Splunk Web, navigate to **Settings > Monitoring Console > Indexing > Performance > Indexing Performance: Instance**.

   The fill ratio of each queue in the Splunk Enterprise Data Pipeline should be at 0% or near zero.

2. Verify the apps are located in the **SPLUNK_HOME/etc/deployment-apps** directory on the deployment server.

   You should have two directories; **hf_base** and **uf_base**.

3. Remote SSH to your heavy forwarder (**10.0.0.77**), and verify that your heavy forwarder is polling your deployment server:

   ~/splunk/bin/splunk show deploy-poll

   If you need to reset the URI, run:

   ~/splunk/bin/splunk set deploy-poll 10.0.0.2##:8089
   ~/splunk/bin/splunk restart

4. From your heavy forwarder (**10.0.0.77**), verify the correct port is enabled with your student id: **~/splunk/bin/splunk display listen** (the output should be **99##**).

   If you need to reset the port, run:

   ~/splunk/bin/splunk enable listen 99##
   ~/splunk/bin/splunk restart

5. Remote SSH into your UF2 (**10.0.0.100**) and verify your forwarder is polling your deployment server:

   ~/splunkforwarder/bin/splunk show deploy-poll

   If you need to reset the URI, run:

   ~/splunkforwarder/bin/splunk set deploy-poll 10.0.0.2##:8089

   ~/splunkforwarder/bin/splunk restart

6. Verify the forwarding destination and receiving host ports are configured correctly and are active for every Splunk component.

   From UF2, run: ./**splunk list forward-server**

   Verify the heavy forwarder (**10.0.0.77**) is listed under Configured but inactive forwards, then restart the forwarder.

   From the heavy forwarder run: **./splunk list forward-server**
   Verify the indexers (**10.0.0.88** and **10.0.0.99)** are listed under **Configured but inactive** forwarders, then restart the forwarder.

   If you see any mistakes, edit the **outputs.conf** file under **SPLUNK_HOME/etc/deployment-apps/[uf_base|hf_base]/local/** on the deployment server and re-deploy the app.

7. Check **splunkd.log** on the forwarder for any recent error or warnings (typically within five minutes).

   cat ~/splunkforwarder/var/log/splunk/splunkd.log | grep 'ERROR\|WARN'

   Or, **egrep 'ERROR|WARN' ~/splunkforwarder/var/log/splunk/splunkd.log**

8. If you still don't get results, ask your instructor for help.

Module 6 Lab – Monitor Inputs

Description

In this lab exercise, you will create all local indexes on the deployment/test server required for subsequent lab exercises. You will test a local directory monitor input on your deployment server/test server. After confirming the events are indexed into the **`test`** index on the test server, you will use the **Add Data** wizard to index the same directory located on UF2 and deploy the input to the **`test`** index located on the remote production indexers. Finally, you will manually edit the attributes of the **`inputs.conf`** to construct a production-ready input and re-index all of the data properly in your production index.

Steps

Task 1: Create production indexes on your deployment/test server.

1. Access Splunk Web on the deployment/test server (**`https://{DS-dns}:8000`**).
2. Click **Settings > Indexes**.
3. Click **New Index**.
4. Populate the form as follows:

   Index Name: **`itops`**
   App: **`Search & Reporting`**
   (This saves the configurations within the Search app-context).

   Notice the default **Index Data Type** is **`Events`**. Leave the rest of the fields empty to accept defaults.
5. Click **Save**.
6. Repeat steps 1 through 5 to create the following indexes:
   - **`sales`**
   - **`securityops`**
   - **`websales`**

Task 2: Add a test directory monitor input to an index on the Deployment Server.

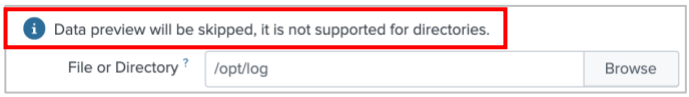
In this task, you will test a local input directory monitor input to index selective directories on the forwarder in bulk. You will use the whitelist and blacklist attributes to define and limit which files are indexed.

7. In Splunk Web on your deployment server, click **Settings > Add Data > Monitor**.
8. On the **Select Source** step, click **Files & Directories**.
9. Click **Browse**, navigate to the directory below and click **Select**.

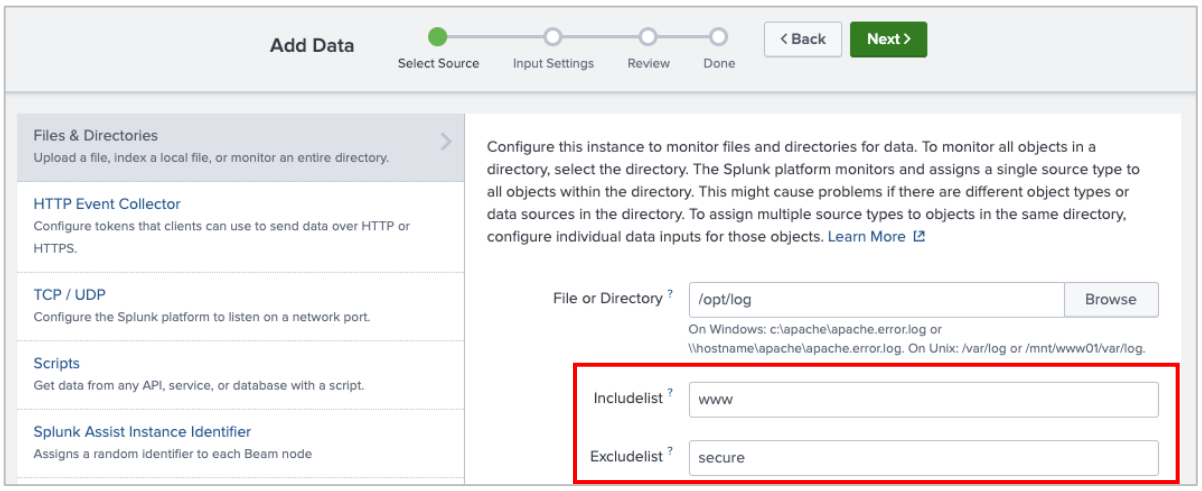
   `/opt/log`

   `C:\opt\log`

   Notice the information icon indicating that data preview will be skipped for directories.



10. On the **Select Source** step, type **`www`** for the **Includelist** and **`secure`** for **Excludelist** and then click **Next**.



11. On the **Input Settings** step, select the following options and click **Review**:

Sourcetype:	**`Automatic`**	
App Context:	**`Search & Reporting (search)`**	
Host field value:	**`splunk##`**	(## should match your student ID)
Index:	**`test`**	

12. Verify the settings on the **Review** step match the following:

Input Type	**`Directory Monitor`**	
Source Path	**`/opt/log`**	(Linux Server)
	`C:\opt\log`	(Windows Server)
Includelist	**`www`**	
Excludelist	**`secure`**	
Source Type	**`Automatic`**	
App Context	**`search`**	
Host	**`splunk##`**	
Index	**`test`**	

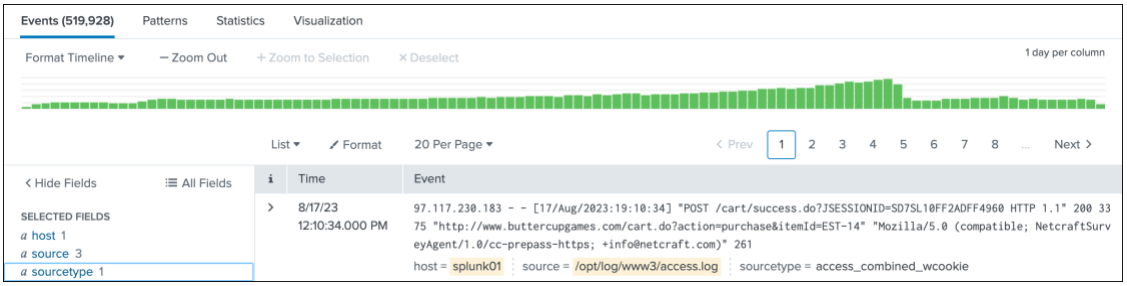
13. Click **Submit**.
14. To verify your monitor input, click **Start Searching**.

    If you get a Welcome message, click **Skip**.

15. Observe the search string (observing that ## should match your student ID):

```
source="/opt/log/*" host="splunk##" index="test"
```

```
source="C:\\opt\\log\\*" host="splunk##" index="test"
```



16. Observe the automatically extracted field names. In the fields sidebar, click the **host**, **source**, and **sourcetype** fields.

    You should see the following field values:

host:	**splunk##**
source (3 total):	**/opt/log/www1/access.log**
	/opt/log/www2/access.log
	/opt/log/www3/access.log
sourcetype:	**access_combined_wcookie**

17. From your deployment server, view the **inputs.conf** file and verify the new stanzas.

```
more /opt/splunk/etc/apps/search/local/inputs.conf
```

```
[monitor:///opt/log/www2/access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie

[splunktcp://9997]
connection_host = ip

[monitor:///opt/log]
blacklist = secure
disabled = false
index = test
whitelist = www
```

```
more "C:\Program Files\Splunk\etc\apps\search\local\inputs.conf"
```

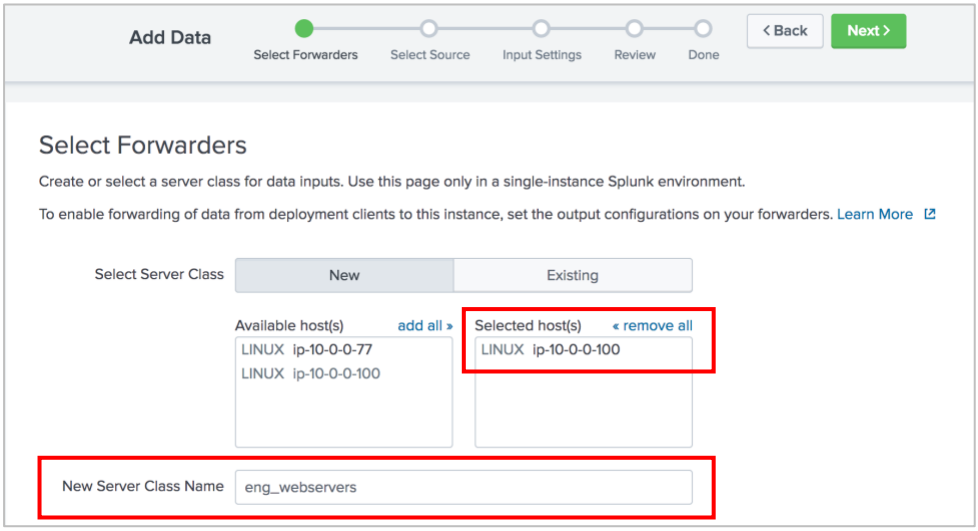
```
[monitor://C:\opt\log\www2\access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie

[splunktcp://9997]
connection_host = ip

[monitor://C:\opt\log]
blacklist = secure
disabled = false
index = test
whitelist = www
```

Task 3: Add a directory monitor input to index remote data from UF2.

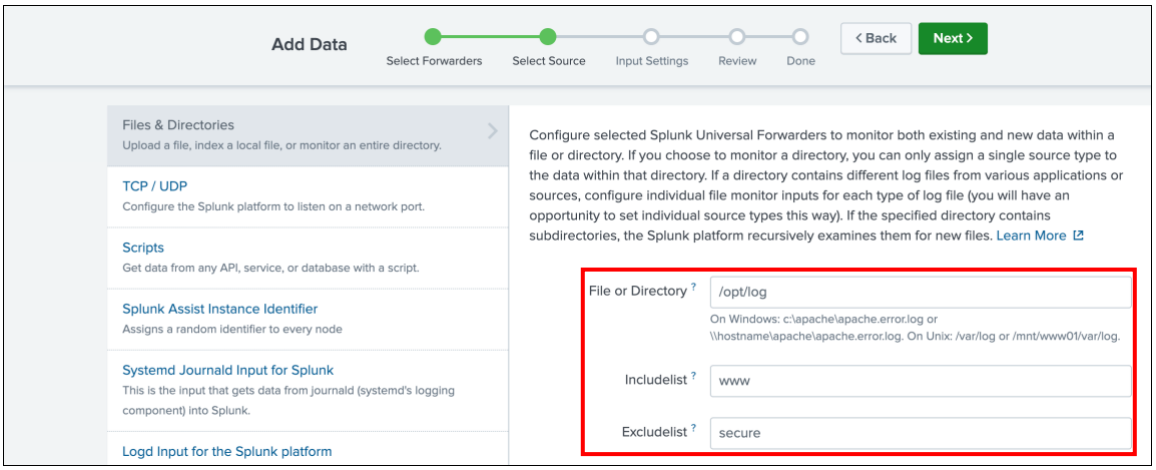
Now that you have successfully indexed a directory monitor input on your test server, you will index the same directories located on UF2 to the remote indexes located on indexers (IDX1 and IDX2). The **Add Data** wizard's **forward** feature automatically creates the **`inputs.conf`** file in a deployable app on the deployment server. It then automatically deploys the app to the forwarder(s) you select on the first page of the wizard.

NOTE: Windows users are still using Linux forwarders. Use the Linux path file as indicated in the input specifications.

18. In Splunk Web on your deployment server, click **Settings > Add Data > Forward**.
19. On the **Select Forwarders** step, configure the form as follows and click **Next**:
    - Select Server Class: **`New`**
    - Selected host(s): **`LINUX ip-10-0-0-100`**
    - New Server Class Name: **`eng_webservers`**



20. On the **Select Source** step, click **Files & Directories** and configure the form as follows, and click **Next**:
    - File or Directory: **`/opt/log`**
    - Includelist: **`www`**
    - Excludelist: **`secure`**



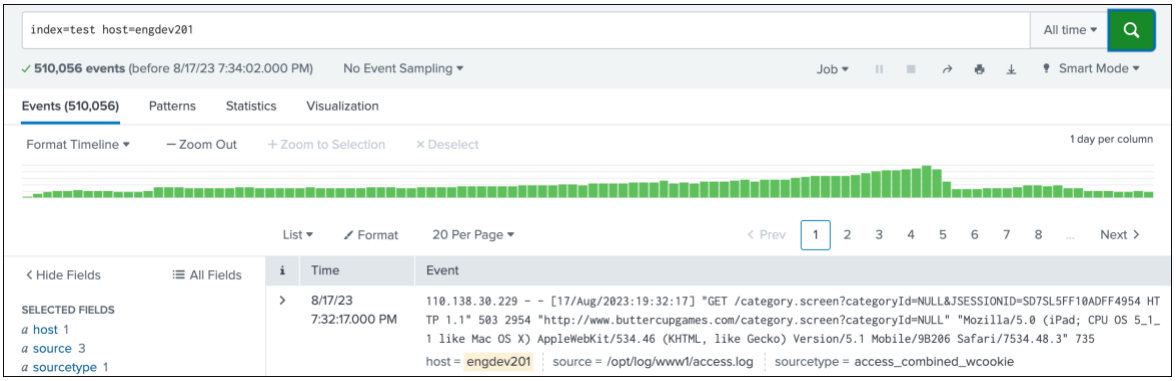
21. For the **Input Settings**, leave the **Source type** as **Automatic** and select **`test`** for the **Index** and click **Review**.
22. Verify your settings match the following, then click **Submit**:

Server Class Name	**`eng_webservers`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log`**
Includelist	**`www`**
Excludelist	**`secure`**
Sourcetype	**`Automatic`**
Index	**`test`**

NOTE: Do not click **Start Searching**! Remember, you just deployed this input to your second forwarder and in the previous lab exercise, you deployed an **`outputs.conf`** file to that forwarder telling it to send all of its data directly to the indexers. Therefore, if you search for the data on your local instance (deployment server), you will see the local data you indexed in Task 1, not the data from the universal forwarder.

23. Open Splunk Web on the search head. Replace the ## with your student ID and execute the following search over **All Time**:

```
index=test host=engdev2##
```

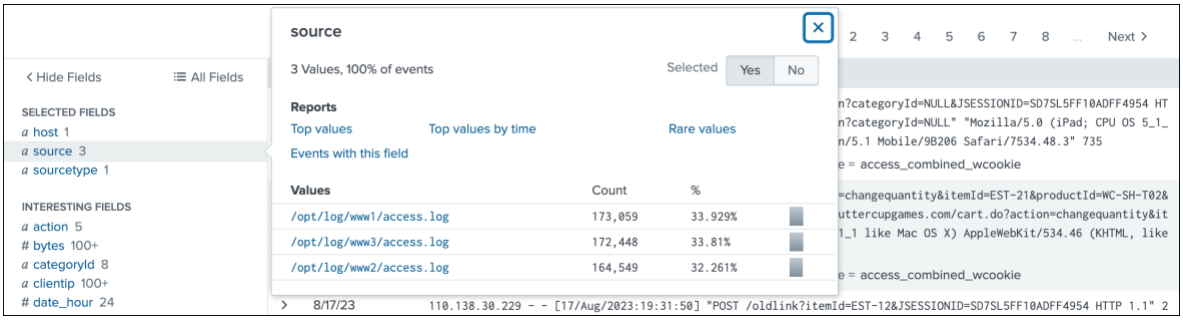


NOTE: It may take a minute before you see results. If no results are found, wait a minute and try again. If the search continues to not show results even after waiting a few minutes, review the Troubleshooting Suggestions section.

24. In the fields sidebar, click the **host**, **source**, and **sourcetype** fields. You should see the following field values:

host:	**engdev2##**
source (3 total):	**/opt/log/www1/access.log** **/opt/log/www2/access.log** **/opt/log/www3/access.log**
sourcetype:	**access_combined_wcookie**

NOTE: It may take several minutes before you see results from all three sources. If your search results match the output above, then you can move on to the next task. If no results are found, wait a minute and try again. If the search continues to not show all 3 sources even after waiting several minutes, review the Troubleshooting Suggestions section.



Task 4: Customize the inputs.conf file manually and re-index to the sales index.

The test run shows that the **host** value is set to the **default-hostname** of the forwarder. The **Add Data** wizard does not provide alternate ways to set the **host** name when adding a remote directory monitor.

You will manually edit the app's **`inputs.conf`** on the deployment server to change the **host** and **index** values. You will then re-deploy the updated input to the forwarder. After the updates, any new data is sent to the new

index, but previously indexed data is not automatically re-indexed. You will have to manually reset the file checkpoints on the forwarder to force all of the data to be re-transmitted.

25. From your deployment server, open the **inputs.conf** file (created by the **Add Data** wizard in Task 1) with a text editor located in the following directory:

```
/opt/splunk/etc/deployment-apps/_server_app_eng_webservers/local/inputs.conf
```

```
C:\Program Files\Splunk\etc\deployment-apps\_server_app_eng_webservers\local\inputs.conf
```

NOTE: The directory **/opt/splunk/etc/deployment-apps/_server_app_eng_webservers** is created by the deployment server in response to creating the **eng_webservers** server class in Task 3 of this lab. It contains configuration files related to the server class.

26. Edit and save the monitor stanza as follows: (Windows users, be sure to close the file after the edit.)

```
[monitor:///opt/log]
blacklist = secure
disabled = false
index = sales                (Update)
whitelist = www
host = www-##                (Add and replace ## with your student ID)
```

NOTE: Any time you update Splunk configuration files in a deployable app at the filesystem-level, the deployment server doesn't know the files have changed, so it doesn't update the checksum value it uses to compare the version of the app on the server with the version on the client. The **reload deploy-server** command causes the deployment server to re-cache the deployable apps and updates the checksum values for any apps that have changed since the last re-cache without having to restart the deployment server. The next time the client phones home, the checksum values of the app will be different, causing the app to be re-deployed.

27. To re-deploy the new **inputs.conf** settings, run this command.

    Splunk will prompt you for the **admin** username and password.

```
/opt/splunk/bin/splunk reload deploy-server
```

```
"C:\Program Files\Splunk\bin\splunk" reload deploy-server
```

28. From your deployment/test server, connect to UF2 (**10.0.0.100**):

After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.100
```

Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session. Log in using your assigned password.

29. Verify the update has been deployed.

```
more ~/splunkforwarder/etc/apps/_server_app_eng_webservers/local/inputs.conf
```

```
[monitor:///opt/log]
blacklist = secure
disabled = false
index = sales
whitelist = www
host = www-##
```
(where ## is your student ID)

NOTE: Because the forwarder has already sent this data once, only new log entries are indexed using the new settings.

30. Trigger the re-indexing of the data in the **sales** index by resetting the monitor checkpoints on the forwarder. The supported method is to stop Splunk, use **btprobe** to reset each monitored input, restart Splunk, and then exit the forwarder's console.

```
cd ~/splunkforwarder/bin

./splunk stop

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/www1/access.log --reset

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/www2/access.log --reset

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/www3/access.log --reset

./splunk start

exit
```

Note

These **btprobe** commands should each be typed all on one line.

NOTE: An alternative method of resetting *all* checkpoints monitored on this instance is by removing the **fishbucket** folder and restarting Splunk. This is simpler method, but also dangerous. Note that this affects *all* monitored inputs, however because we are on a test server where we want to reset all checkpoints, this may not be a concern. One side effect is licensing use of re-indexing the data. Another concern is accidental deletion of an incorrect folder, which can be irreparable.

```
~/splunkforwarder/bin/splunk stop

cd ~/splunkforwarder/var/lib/splunk/
rm -r fishbucket

~/splunkforwarder/bin/splunk start

exit
```

Warning

Do not run these commands on your instance unless instructed to by your instructor.

NOTE: Another method for resetting all monitored inputs is running the **`splunk clean eventdata -index _thefishbucket`** command and restarting Splunk *on the indexer*. This command should be used with caution, as typos or running on incorrect instances can have disastrous consequences: Running **`splunk clean`** without the **`-index`** option will remove ***all*** indexes from that Splunk instance. (Students do not have permissions to run this command on the indexer in this lab environment.)

```
cd <SPLUNK_HOME>/bin
./splunk stop

./splunk clean eventdata -index _thefishbucket

./splunk start
```

Warning

Do not run these commands in this lab environment.

Check Your Work

Task 5: Verify your forwarder is sending the events to the indexer.

31. In Splunk Web on the search head execute the following search over the **Last 4 hours** (replace the ## with your student ID):

    `index=sales host=www-##`

    Eventually, you should see one sourcetype and three sources in your search results. It may take several minutes before you see all 3 sources.

Troubleshooting Suggestions

1. On your deployment server, navigate to the **Settings > Forwarder management** page and click the **Clients** tab.

   Verify your client is still phoning home and has reported 2 deployed apps.

2. Remote SSH into UF2 (**`10.0.0.100`**) and confirm the deployed input stanza:

   1st phase deployment:

   ```
   more ~/splunkforwarder/etc/apps/_server_app_eng_webservers/local/inputs.conf
   [monitor:///opt/log]
   blacklist = secure
   disabled = false
   whitelist = www
   index = test
   ```

   2nd phase deployment:

   ```
   more ~/splunkforwarder/etc/apps/_server_app_eng_webservers/local/inputs.conf
   [monitor:///opt/log]
   blacklist = secure
   disabled = false
   ```

```
index = sales
whitelist = www
host = www-##
```
(where ## is your student ID)

3. If you need to make changes, edit the **inputs.conf** file on the deployment server, reset the monitor checkpoints on the forwarder, and close the remote SSH session. Otherwise

```
cd ~/splunkforwarder/var/lib/splunk/
rm -r fishbucket
~/splunkforwarder/bin/splunk restart
exit
```

If you still don't get results, ask your instructor for help.

Module 7 Lab – Network Inputs

Description

Your instructor has configured a source to send TCP traffic to your UF2 (**`10.0.0.100`**). You will deploy a network input to the UF2 (**`10.0.0.100`**) which will only receive events from a known host and forward that data to the indexers.

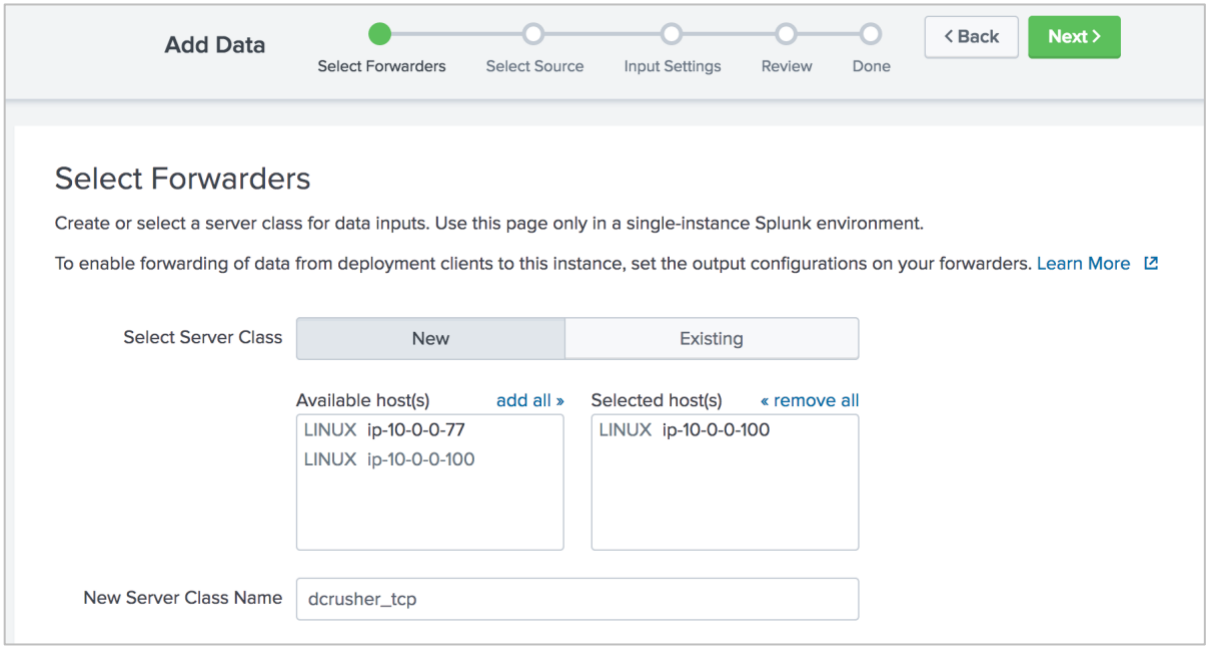
Steps

Task 1: Add a forward network input and deploy it to UF2 (10.0.0.100).

To examine TCP data coming to UF2 (**`10.0.0.100`**), index TCP events into the **test** index by deploying a remote network input.

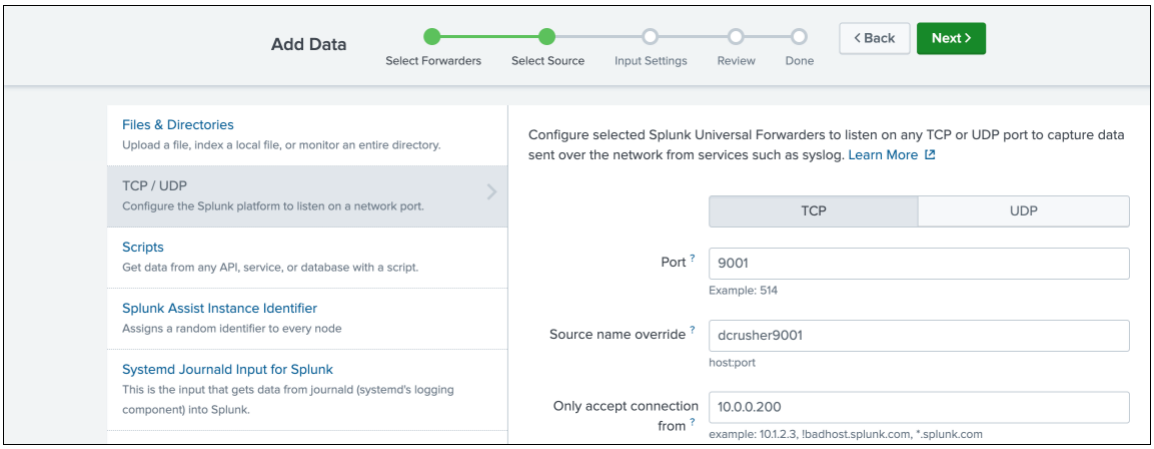
1. In Splunk Web on your deployment server, click **Settings > Add Data > Forward**.
2. On the **Select Forwarders** step, configure the form as follows, and then click **Next**:

Select Server Class:	**`New`**
Selected host(s):	**`LINUX ip-10-0-0-100`**
New Server Class Name:	**`dcrusher_tcp`**

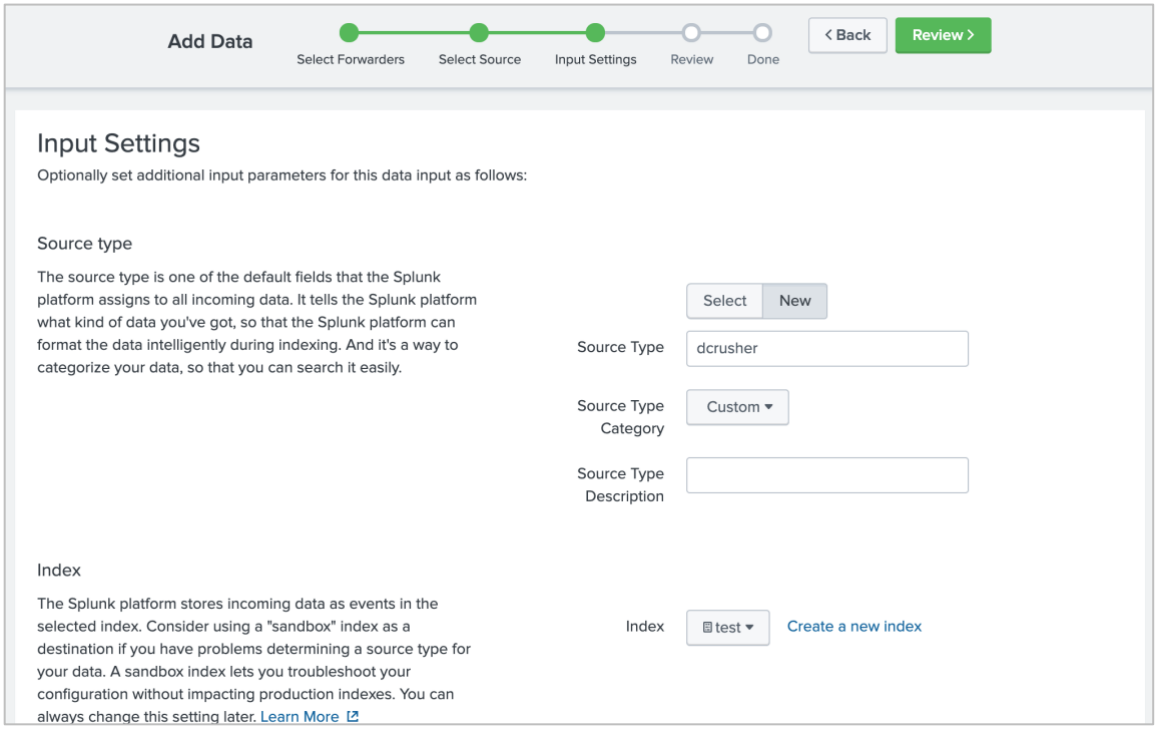


3. On the **Select Source** step, click **TCP / UDP** and configure the form as follows (replace ## with your student ID), and then click **Next**:

Select	**`TCP`**	
Port:	**`90##`**	(where ## is your student ID)
Source name override:	**`dcrusher90##`**	(where ## is your student ID)
Only accept connection from:	**`10.0.0.200`**	



4. For the **Input Settings**, select **New**, enter **Source type** as **`dcrusher`** and select **`test`** for the **Index**.



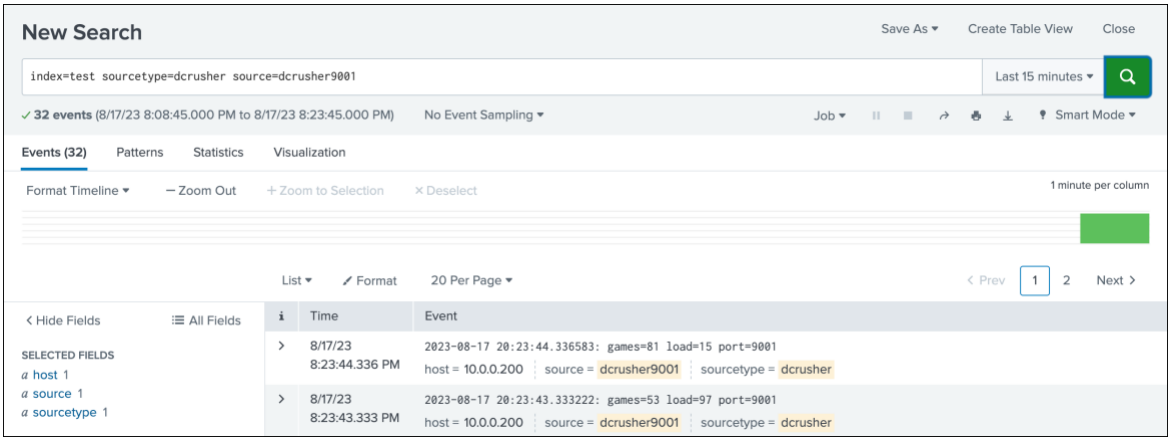
5. Click **Review** and make sure the input settings match the following:

Server Class Name	**`dcrusher_tcp`**	
List of Forwarders	**`LINUX|ip-10-0-0-100`**	
Input Type	**`TCP Port`**	
Port Number	**`90##`**	(where ## is your student ID)
Source name override	**`dcrusher90##`**	(where ## is your student ID)
Restrict to Host	**`10.0.0.200`**	
Source Type	**`dcrusher`**	
Index	**`test`**	

6. Click **Submit**.

7. In Splunk Web on the search head execute the following search over the **Last 15 minutes** (replace the ## with your student ID):

   `index=test sourcetype=dcrusher source=dcrusher90##`



NOTE: You may need to wait several minutes to see results. If you are not seeing any results, ensure you are performing the search on the search head, and not the deployment server.

8. In the fields sidebar, click the **host**, **source** and **sourcetype** fields. You should see the following field values:

host:	**`10.0.0.200`**
source:	**`dcrusher90##`**
sourcetype:	**`dcrusher`**

   The test run shows that the IP address of the sender is used to set the value of the **`host`** field. If the test worked, then move on to the next task.

Task 2: Modify the host and index values, then finalize it as a production input.

In this task, you manually edit the **`inputs.conf`** file to set the host value to **dcrusher_devserver** and route the data to the **`itops`** index.

9. From your deployment server, open the **`inputs.conf`** file with a text editor:

   ```
   /opt/splunk/etc/deployment-apps/_server_app_dcrusher_tcp/local/inputs.conf
   ```

   ```
   C:\Program Files\Splunk\etc\deployment-apps\_server_app_dcrusher_tcp\local\inputs.conf
   ```

10. Edit and save the input stanza as follows *(where ## is your student ID)*:

    ```
    [tcp://10.0.0.200:90##]
    connection_host = none          (Change)
    host = dcrusher_devserver       (Add)
    index = itops                   (Change)
    source = dcrusher90##
    sourcetype = dcrusher
    ```

NOTE: Windows users, be sure to close the file after the edit.

11. To re-deploy the modified input, run: You may be prompted to login as **admin**.

```
/opt/splunk/bin/splunk reload deploy-server
```

```
"C:\Program Files\Splunk\bin\splunk" reload deploy-server
```

Check Your Work

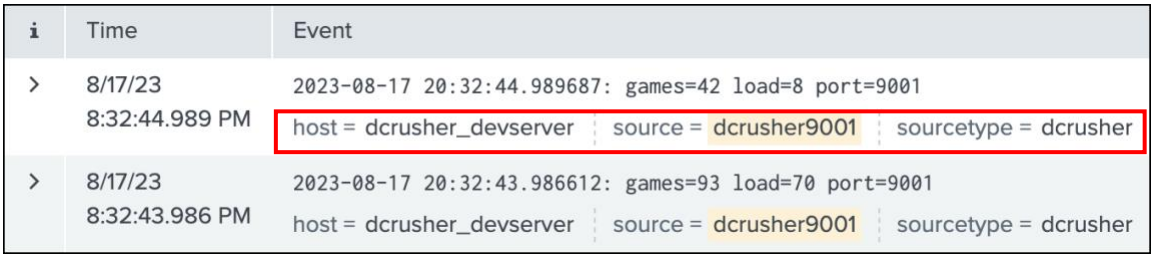
Task 3: Verify the forwarded TCP input events.

12. In Splunk Web on the search head execute the following search over the **Last 15 minutes** (replace the ## with your student ID):

```
index=itops source=dcrusher90##
```

You should see the following field values:

host:	**dcrusher_devserver**
source:	**dcrusher90##**
sourcetype:	**dcrusher**



Troubleshooting Suggestions

1. Check **splunkd.log** for any IO related event messages.

   On the forwarder, check (Note that the following commands should be on a single line.)

   Are there any errors?

```
tail ~/splunkforwarder/var/log/splunk/splunkd.log | grep 'ERROR\|WARN'
```

2. Is the TCP port configured? (Use your port number instead of 90##):

```
cat ~/splunkforwarder/var/log/splunk/splunkd.log | grep -E 'TcpInputConfig.*90##'
```

3. Is the forwarder processing the TCP events?

```
cat ~/splunkforwarder/var/log/splunk/splunkd.log | grep -E 'TcpInputProc.*90##'
```

4. On the search head, search the index metrics for the TCP traffic to check for any events received on forwarder #2:

```
index=_internal host=engdev2## component=Metrics name=tcpin_queue
```

```
index=_internal host=engdev2## component=Metrics series="dcrusher*"
```

5. Confirm the deployed input stanza on the forwarder.

```
more ~/splunkforwarder/etc/apps/_server_app_dcrusher_tcp/local/inputs.conf
```

```
[tcp://10.0.0.200:90##]          (where ## is your student ID)
connection_host = none
host = dcrusher_devserver
index = itops
source = dcrusher90##            (where ## is your student ID)
sourcetype = dcrusher
```

If you still don't get any results, ask your instructor for help.

Module 8 Lab – Scripted Inputs

Description

The Linux **vmstat** command is a useful tool for gathering a snapshot of server information such as memory usage, processes, and CPU load. Indexing this data in Splunk is useful for trending analysis and capacity planning.

In this lab exercise, you will deploy a scripted input to a Linux forwarder and collect **vmstat** data.

Steps

Task 1: Add a scripted input on your deployment server and deploy it to the forwarder #2.

1. From the deployment server's filesystem, copy the **/opt/scripts/myvmstat.sh** file to the **SPLUNK_HOME/bin/scripts** folder.

```
cp /opt/scripts/myvmstat.sh /opt/splunk/bin/scripts
```

```
copy C:\opt\scripts\myvmstat.sh "C:\Program Files\Splunk\bin\scripts\"
```

(Alternatively use the Windows File Explorer to copy **myvmstat.sh** from **C:\opt\scripts** to **C:\Program Files\Splunk\bin\scripts**)

2. View the script.

```
more /opt/splunk/bin/scripts/myvmstat.sh
```

```
more "C:\Program Files\Splunk\bin\scripts\myvmstat.sh"
```

The script should contain these lines:

```
#!/bin/sh
/usr/bin/vmstat
```

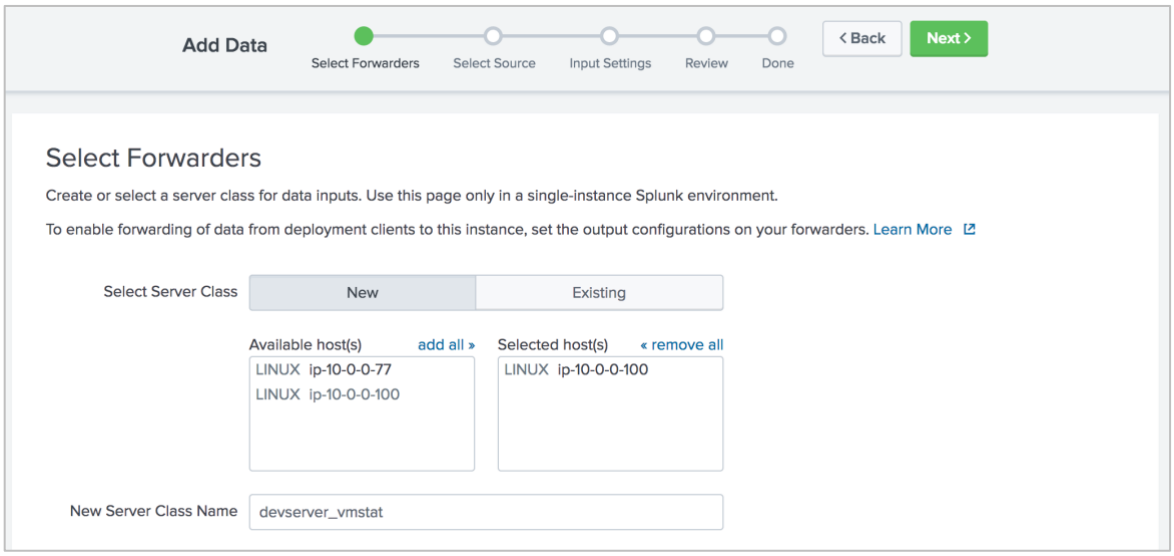
Running the script on a *nix server should produce output that contains information about server memory and CPI usage, and should look similar to this:

```
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0      0 569860 150456 4425276    0    0     0    48    6    2  4  1 95  0  0
```

3. In Splunk Web on your deployment server, click **Settings > Add Data > Forward**.

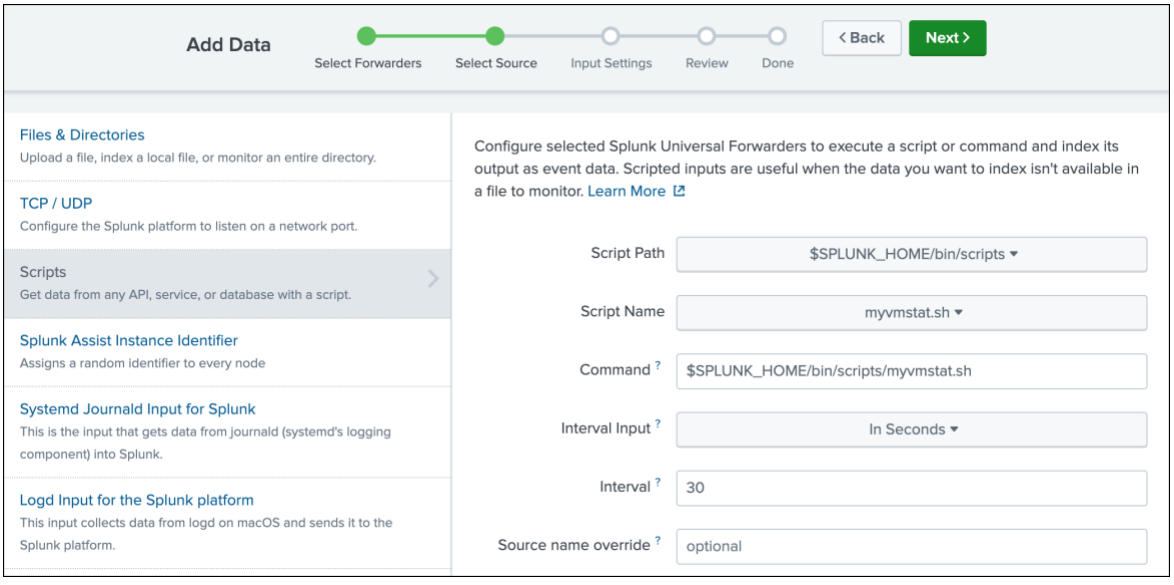
4. On the **Select Forwarders** step, configure the form as follows, and then click **Next**:

   Select Server Class: **New**
   Selected host(s): **LINUX ip-10-0-0-100**
   New Server Class Name: **devserver_vmstat**



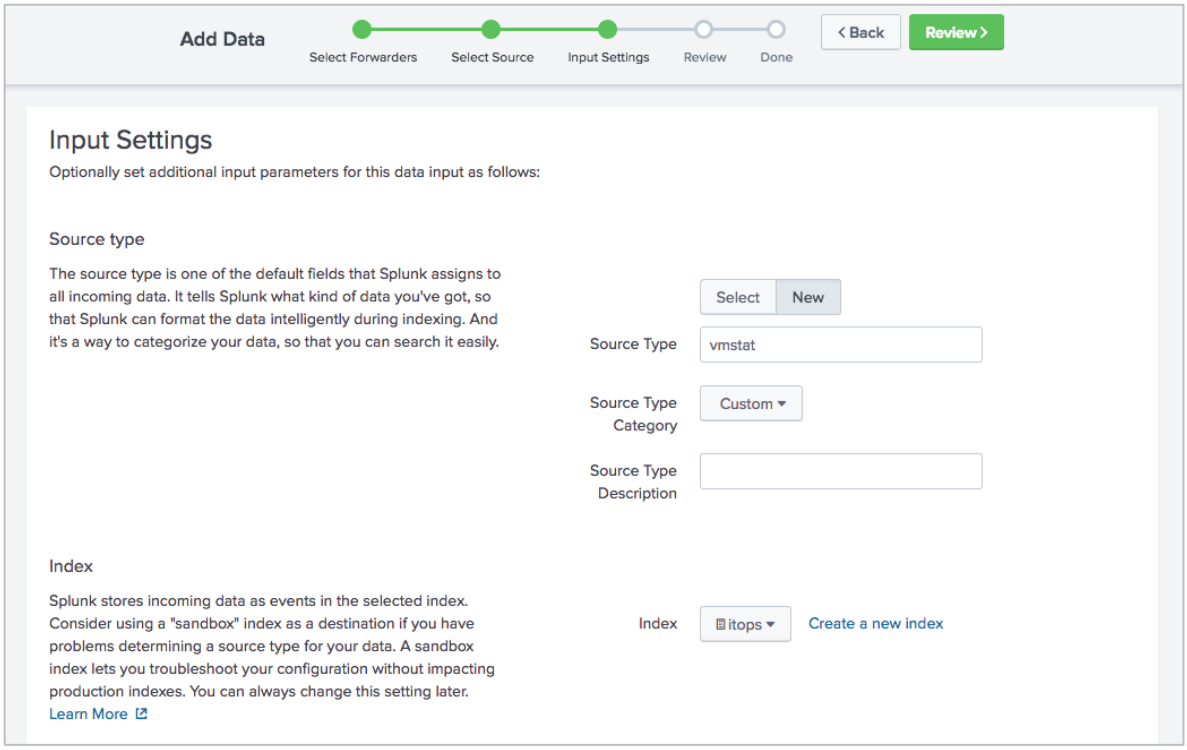
5. On the **Select Source** step, click **Scripts** and configure the form as follows and click **Next**:

   Script Path: **$SPLUNK_HOME/bin/scripts**
   Script Name: **myvmstat.sh**
   Command: **$SPLUNK_HOME/bin/scripts/myvmstat.sh** (this auto populates)
   Interval: **30**



6. For the **Input Settings**, to the right of **Source type** select **New**, and configure the form as follows and click **Review**:

Source type:	**vmstat**
Index:	**itops**



7. Make sure the input settings match the following:

Server Class Name	**devserver_vmstat**
List of Forwarders	**LINUX ip-10-0-0-100**
Input Type	**Script**
Command	**$SPLUNK_HOME/bin/scripts/myvmstat.sh**
Interval	**30**
Source name override	**N/A**
Source type	**vmstat**
Index	**itops**

8. Click **Submit**.
9. For **Windows Students Only:** Remote SSH to forwarder #2 (**SSH to UF2**) and change the file permission of the script:

```
chmod +x ~/splunkforwarder/etc/apps/_server_app_devserver_vmstat/bin/myvmstat.sh
```

NOTE: Windows students must perform the above step every time a scripted input is re-deployed to a Linux forwarder.

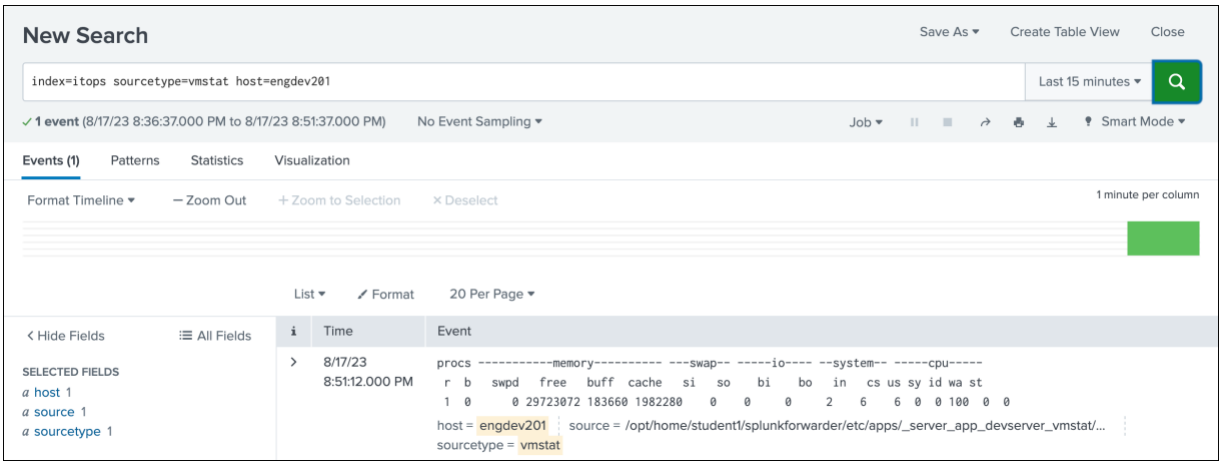
Check Your Work

Task 2: Verify the output of your scripted input.

10. In Splunk Web on the search head execute the following search over the **Last 15 minutes** (replace the ## with your student ID):

    ```
    index=itops sourcetype=vmstat host=engdev2##
    ```

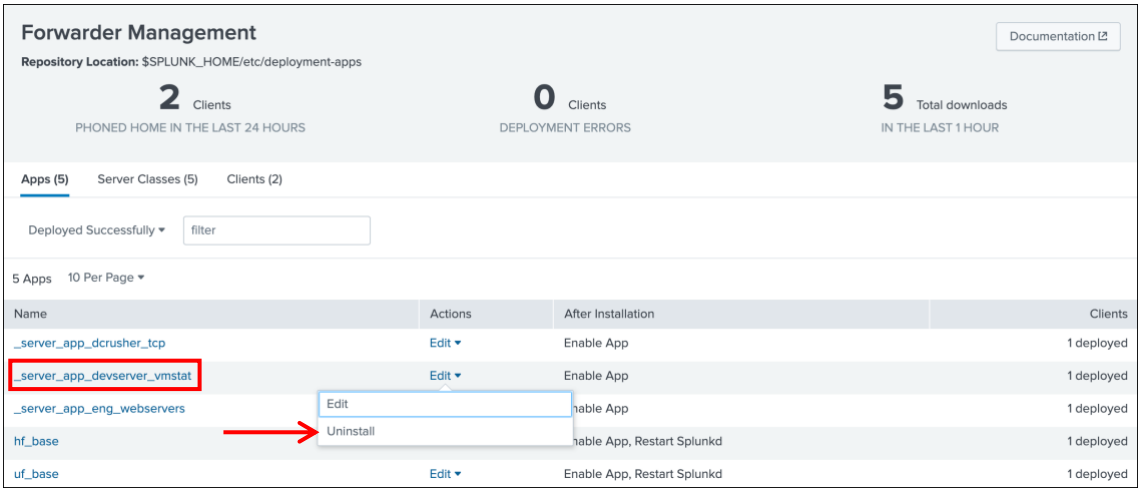
    You may need to wait several minutes to see results. When you do, do not navigate away from these search results.



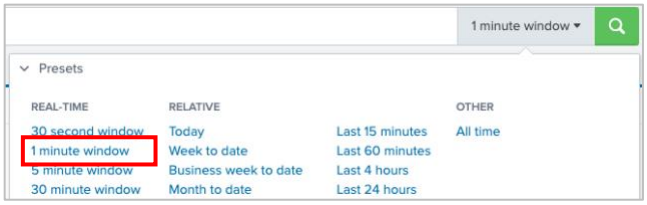
Task 3: Disable the forward scripted input.

After you confirm the scripted input is working, uninstall the deployment app. You are doing this to reduce the system load on the forwarder, as it is a shared host in this lab environment.

11. On the deployment server, navigate to **Settings** > **Forwarder management** and click the **Apps** tab.
12. For the app **`_server_app_devserver_vmstat`**, click **Edit > Uninstall > Uninstall**.



13. Switch back to Splunk Web on the search head and change the time range of the search to: **REAL-TIME > 1 minute window**.



14. Wait until the event count drops to **0** (0 of X events matched) and then stop (click ■) the real-time search.

Troubleshooting Suggestions

If the scripted input is not returning the expected results, troubleshoot by isolating the issue.

1. Verify the syntax and spelling.

   Verify the script name in the **`inputs.conf`** has the full script name including the **`.sh`** extension.

2. Search for forwarder errors in the internal index:

   ```
   index=_internal sourcetype=splunkd component=ExecProcessor host=engdev2##
   ```

3. Test your script on the forwarder and confirm that the script itself is producing some output.

   ```
   ~/splunkforwarder/etc/apps/_server_app_devserver_vmstat/bin/myvmstat.sh
   ```

4. Check for any errors in the splunkd.log on the forwarder #2 for script actions.

   ```
   tail ~/splunkforwarder/var/log/splunk/splunkd.log | grep 'ERROR\|WARN'
   ```

5. Check for any scripted input related **`splunkd`** logs.

   ```
   tail ~/splunkforwarder/var/log/splunk/splunkd.log | grep 'ExecProcessor'
   ```

   An error message ***`bad interpreter`*** in the forwarder's splunkd.log indicates that *nix scripts were drafted using a Windows OS. A file created in a Windows environment may be using a DOS-based carriage return. Check the file format of the **`myvmstat.sh`** file and convert it to a UNIX format.

If you still don't see events on the search head, ask your instructor for help.

Module 9 Lab – Agentless Inputs with HTTP Event Collector

Description

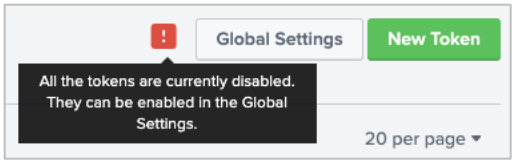
In this lab exercise, you enable and configure the HTTP event collector (HEC) on the deployment/test server. Once configured, you can transmit HTTP data and the deployment/test server will parse the data and forward the resulting events to the local indexers.

Steps

Task 1: Enable HTTP event collector on your HEC Receiver (deployment/test server).

1. In Splunk Web on your deployment server, navigate to **Settings** > **Data inputs**.
2. From **Local inputs**, click **HTTP Event Collector**.

   Notice the red exclamation mark next to the **Global Settings** button. If you float your cursor over it, it states: “All the tokens are currently disabled. They can be enabled in the Global Settings.”



3. Click **Global Settings**.
4. Select the following settings:

All Tokens	Click the **`Enabled`** button
Default Source Type	**`Structured > json_no_timestamp`**
Default Index	**`test`**
Default Output Group	**`None`**
Use Deployment Server	**`off`**
Enable SSL	**`off`** **`(This is on by default)`**
HTTP Port Number	**`8088`**

5. Click **Save**.

   Notice the red exclamation mark next to the **Global Settings** button was removed.

6. Click **New Token**.

   The **Select Source** step of the **Add Data** wizard opens with the **HTTP Event Collector** selected in the left panel.

7. In the **Name** field, type: **`iot_sensors`**. From the **Output Group (optional)**, notice that **None** is selected in the drop-down menu and click **Next**.
8. On the **Input Settings** page, set the values to the following:

Source type	**`Automatic`**
Select Allowed Indexes	Add **`itops`** and **`test`** to the **Selected item(s)**
Default Index	**`test`**

9. Click **Review** and make sure all the settings match:

Input Type	**Token**
Name	**iot_sensors**
Source name override	**N/A**
Description	**N/A**
Enable indexer acknowledgements	**No**
Output Group	**N/A**
Allowed indexes	**itops** and **test**
Default index	**test**
Source Type	**Automatic**
App Context	**search**

10. Click **Submit**.

    The **Token has been created successfully** message displays with the token value of the collector. You will share this token with the developers who will send events to the indexer.

11. Copy the **Token Value** and save it to a text document.

Check Your Work

Task 2: Send test events to your indexer using `hec1.sh` script.

In real practice, developers would create programs or scripts to send events to the receiving collector. In this lab environment, scripts are provided for you. You will run the **`hec1.sh`** script in this task.

12. From your deployment/test server, connect to UF1 (**`10.0.0.50`**):

    After establishing an SSH session to your deployment/test server, use SSH to connect to UF1 (**10.0.0.50**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.50
```

    Open the **PuTTy** application, click on session "**SSH to UF1**" and click **Open** to start the session. Log in using your assigned password.

13. Execute the following **`export`** commands to set the **`H_SERVER`** and **`H_TOKEN`** environment variables for the HEC events on UF1:

```
export H_SERVER=10.0.0.2##                          (where ## is your student ID)
export H_TOKEN=CCCCCCCC-xxxx-yyyy-zzzz-99999999999  (paste the token from your text document)
```

    These variables set the IP address and HTTP token for the upcoming curl commands.

14. Execute the following **`echo`** commands to verify the **`H_SERVER`** and **`H_TOKEN`** environment variables are set correctly:

```
echo $H_SERVER
10.0.0.2##                          (where ## is your student ID)
echo $H_TOKEN
CCCCCCCC-xxxx-yyyy-zzzz-99999999999 (this should match the token you used earlier)
```

15. To send basic Event Collector data, examine and run the **hec1.sh** script in **/opt/scripts**:

```
/opt/scripts/hec1.sh
```

The script uses the following **curl** command to submit the JSON events to your indexer:

```
curl http://${H_SERVER}:8088/services/collector \
-H "Authorization: Splunk ${H_TOKEN}" \
-d ' {"event": "Hello World 1"}'
```

NOTE: If you get the **curl: (6) Could not resolve host** message, make sure **H_SERVER** is set to your deployment server's IP address.

If you get the the **curl: (7) Failed to connect to 10.0.0.2## port 8088:Connection refused** message, verify your HTTP Event Collector global settings.

If the submit is successful, you will get the **{"text":"Success","code":0}** message 10 times.

If it fails, you will see an error message; e.g. **{"text":"Invalid token","code":4}**

16. From your deployment/test server, execute the following search over the **last 15 minutes**, replacing the ## with your student ID:

```
index=test source=http* host=*##:8088
```

IMPORTANT: In many labs we run searches from the Search Head, but in this case our input was set to the **deployment/test server**, so the search is being performed on that Splunk instance.

You should see 10 events for each successful run of the **hec1.sh** script, with the following field values:

host:	**10.0.0.2##:8088**
source:	**http:iot_sensors**
sourcetype:	**json_no_timestamp**

Task 3: Send test events to your indexer using hec2.sh script.

In real practice, developers would create programs or scripts to send events to the receiving collector. In this lab environment, scripts are provided for you. You will run the **hec2.sh** script in this task.

17. Return to UF1 (**10.0.0.50**)
18. Execute the following **echo** commands to verify the **H_SERVER** and **H_TOKEN** environment variables are set correctly:

```
echo $H_SERVER
10.0.0.2##                              (where ## is your student ID)
echo $H_TOKEN
CCCCCCCC-xxxx-yyyy-zzzz-99999999999     (this should match the token you used earlier)
```

If the variables are not set correctly and returning these values, perform step 13 of this lab again.

19. To send another set of events that override the default metadata, run the **hec2.sh** script and, when prompted to enter a message, type your two-digit student ID followed by a personalized message.

```
/opt/scripts/hec2.sh
This script will send 10 Http Collector events and override the default metadata.
Enter a short message?
{student ID} YOUR PERSONALIZED MESSAGE

About to send HEC events with your message {student ID} YOUR PERSONALIZED MESSAGE to index
itops...Press 'y' to continue or any to abort:y
y
{"text":"Success","code":0}
...
```

NOTE: The message must begin with your student ID in order to validate the data. For example if your student ID is **03**, you could type "**03 Splunk rules!**"

The **hec2.sh** script uses the following **curl** command to override the default metadata:

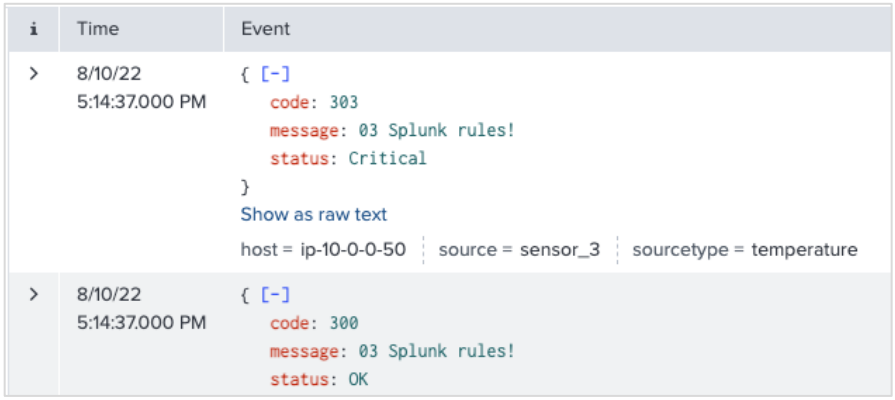
```
curl http://${H_SERVER}:8088/services/collector \
-H "Authorization: Splunk ${H_TOKEN}" \
-d '{"index":"'"$index"'", "host":"'"$HOSTNAME"'", "sourcetype":"'"$sourcetype"'",
source":"'"$source"'", "event":{"code":"'"$code"'", "status":"'"$status"'",
"message":"'"$msg"'"}}'
```

20. Close your remote SSH session.

```
exit
```

21. From your deployment/test server, execute the following search over the **last 15 minutes**, replacing the ## with your student ID:

```
index=itops message="##*"
```



Troubleshooting Suggestions

If you get the error message, "**`curl: (56) Recv failure: Connection reset by peer`**", it means you did NOT uncheck the **Enable SSL** box in the **Global Settings** (Step 4).

1. Confirm the resulting input stanzas on the deployment server:

```
more SPLUNK_HOME/etc/apps/splunk_httpinput/local/inputs.conf
```

```
[http]
disabled = 0
enableSSL = 0
index = test
sourcetype = json_no_timestamp
```

```
more SPLUNK_HOME/etc/apps/search/local/inputs.conf
```

```
...
[http://iot_sensors]
disabled = 0
index = test
indexes = itops,test
token = <generated_token>
```

2. Use the **btool** command with the **`--debug`** argument to display the **`iot_sensor inputs.conf`** stanzas.

```
./splunk btool inputs list http://iot_sensors --debug
```

```
/opt/splunk/etc/apps/search/local/inputs.conf [http://iot_sensors]
/opt/splunk/etc/system/default/inputs.conf    _rcvbuf = 1572864
/opt/splunk/etc/apps/search/local/inputs.conf disabled = 0
/opt/splunk/etc/system/local/inputs.conf      host = splunk##
/opt/splunk/etc/apps/search/local/inputs.conf index = test
/opt/splunk/etc/apps/search/local/inputs.conf indexes = itops,test
/opt/splunk/etc/apps/search/local/inputs.conf token = <generated_token>
```

Module 11 Lab – Fine-tuning Inputs

Description

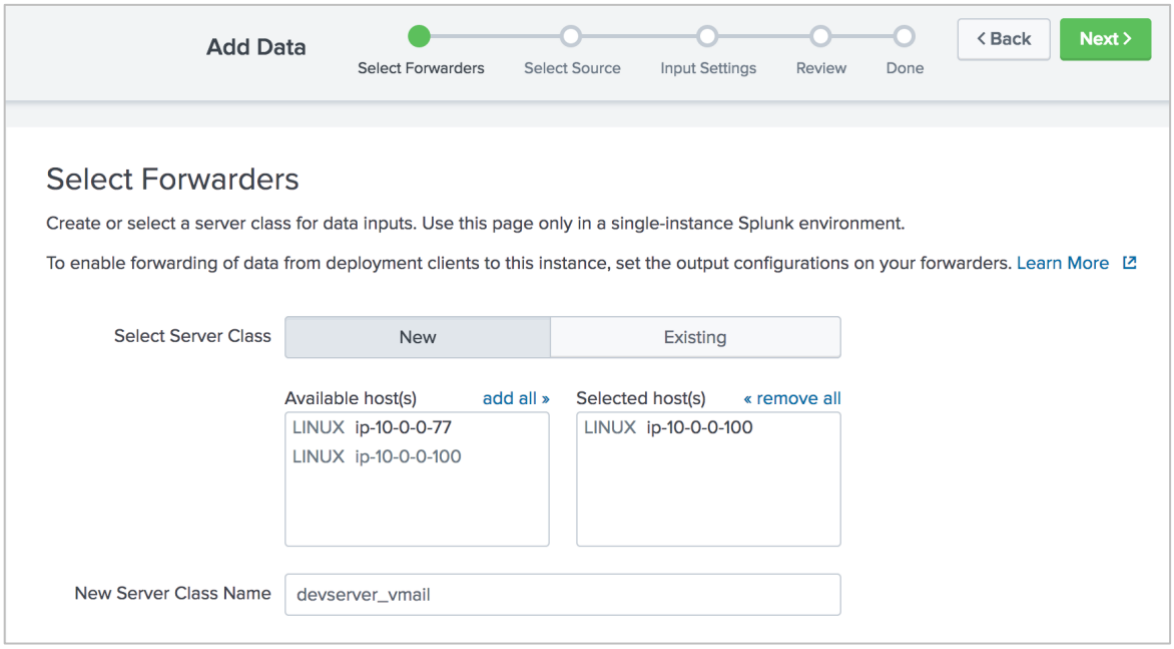
In this lab exercise, you add a remote directory monitor input to index several sources on UF2 using the automatic source typing feature. While this is a convenient feature, Splunk does not always assign the correct sourcetype for every file in a directory. When this happens, you must intervene to override the sourcetype.

Steps

Task 1: Add a remote test directory monitor input to sample the auto-sourcetype behavior.

1. In Splunk Web on your deployment server, click **Settings > Add Data > Forward**.
2. On the **Select Forwarders** step, configure the form as follows then click **Next**:

Select Server Class:	**`New`**
Selected host(s):	**`LINUX ip-10-0-0-100`**
New Server Class Name:	**`devserver_vmail`**



3. On the **Select Source** step, click **Files & Directories** and configure the **File or Directory** to **`/opt/log/vmail`**, and click **Next**.
4. For the **Input Settings**, leave the **Source type** to **Automatic**, select the **`test`** index, and click **Review**.
5. Verify your input matches the following and click **Submit**:

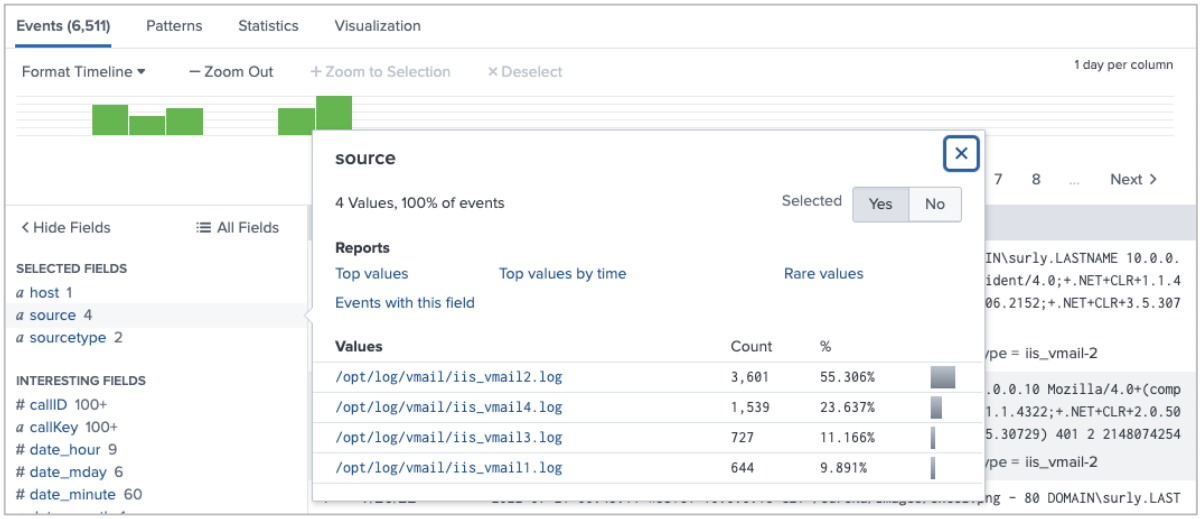
Server Class Name	**`devserver_vmail`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log/vmail`**
Includelist	**`N/A`**
Excludelist	**`N/A`**
Source Type	**`Automatic`**
Index	**`test`**

6. In Splunk Web on the search head execute the following search over the **Last 30 days** (replace the ## with your student ID):

   ```
   index=test source=*vmail* host=engdev2##
   ```

   You should see the following field values:

host:	**engdev2##**
source (4 total):	**/opt/log/vmail/iis_vmail1.log** **/opt/log/vmail/iis_vmail2.log** **/opt/log/vmail/iis_vmail3.log** **/opt/log/vmail/iis_vmail4.log**
Sourcetype (2 total):	**iis_vmail** **iis_vmail-2**



NOTE: If you are not seeing any results, ensure you are on the search head, and searching over the **Last 30 days**.

Task 2: Override the sourcetype of iis_vmail3.log.

In this task, you create a **props.conf** file in the **deployment-apps** directory and deploy it to your universal forwarder UF2. This file does not currently exist. You also edit the directory input to re-send the data to the **itops** index. Because the data has already been transmitted, you will use the **btprobe** command to reset the file checkpoints for two of the log files.

7. From your deployment server, use a text editor to create a new **props.conf** file at:

   ```
   /opt/splunk/etc/deployment-apps/_server_app_devserver_vmail/local/props.conf
   ```

   ```
   C:\Program Files\Splunk\etc\deployment-apps\_server_app_devserver_vmail\local\props.conf
   ```

8. Insert the following text:

   ```
   [source::/opt/log/vmail/iis_vmail3.log]
   sourcetype = acme_voip
   ```

9. Save and close the file.

10. Using a text editor, open the **inputs.conf** file for the **vmail** directory input.

```
/opt/splunk/etc/deployment-apps/_server_app_devserver_vmail/local/inputs.conf
```

```
C:\Program Files\Splunk\etc\deployment-apps\_server_app_devserver_vmail\local\inputs.conf
```

11. Change the **vmail** directory input's index attribute as follows:

```
[monitor:///opt/log/vmail]
disabled = false
index = itops                    (Change)
```

12. Save and close the file.
13. To re-deploy the modified input, run the following command.

    Splunk may prompt you for the **admin** username and password.

```
/opt/splunk/bin/splunk reload deploy-server
```

```
"C:\Program Files\Splunk\bin\splunk" reload deploy-server
```

NOTE: You are deploying the **props.conf** and the **inputs.conf** updates to UF2. Data is not parsed on the universal forwarder; the source type override functionality is an input phase activity. Later, you will deploy **props.conf** to the heavy forwarder to parse data prior to sending the data to the indexers.

14. From your deployment/test server, connect to UF2 (**10.0.0.100**):

After establishing an SSH session to your deployment/test server, use SSH to connect to UF2 (**10.0.0.100**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.100
```

Open the **PuTTy** application, click on session "**SSH to UF2**" and click **Open** to start the session. Log in using your assigned password.

15. Verify the update was deployed.

```
cat ~/splunkforwarder/etc/apps/_server_app_devserver_vmail/local/inputs.conf
[monitor:///opt/log/vmail]
disabled = false
index = itops

cat ~/splunkforwarder/etc/apps/_server_app_devserver_vmail/local/props.conf
[source::/opt/log/vmail/iis_vmail3.log]
sourcetype = acme_voip
```

16. To trigger the re-indexing of the same sources on the forwarder, reset the individual checkpoints for two of the **iis_vmail** logs on UF2 (**10.0.0.100**) by running the following commands.

```
cd ~/splunkforwarder/bin

./splunk stop
```

```
./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/vmail/iis_vmail2.log --reset
...
Record (key 0x...) reset.

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/vmail/iis_vmail3.log --reset

...
Record (key 0x...) reset.

./splunk start
```

Note

These **btprobe** commands should each be typed all on one line.

17. Exit UF2.

```
exit
```

Check Your Work

Task 3: Verify the source type.

18. In Splunk Web on the search head execute the following search over the **Last 30 days** (replace the ## with your student ID):

```
index=itops source=*vmail* host=engdev2## | stats count by source, sourcetype
```

source	sourcetype	count
/opt/log/vmail/iis_vmail2.log	iis_vmail-2	3601
/opt/log/vmail/iis_vmail3.log	acme_voip	727

19. Confirm that the **itops** index contains only the sources for the two files where we reset the checkpoints (**iis_vmail2.log** and **iis_vmail3.log**). Also confirm that **iis_vmail3.log** is now using the overridden sourcetype **acme_voip**, while **iis_vmail2.log** is still using the automatic sourcetype values of **iis_vmail** or **iis_vmail-2**.

Troubleshooting Suggestions

If the configuration is not producing the expected results, check your configurations.

1. Verify the syntax, spelling, and the key values in the configuration files.

```
~/splunkforwarder/bin/splunk btool inputs list monitor:///opt/log/vmail
~/splunkforwarder/bin/splunk btool props list source::/opt/log/vmail/iis_vmail
```

2. Check the splunkd.log on the forwarder for any monitoring process errors.

```
tail ~/splunkforwarder/var/log/splunk/splunkd.log | grep 'TailingProcessor'
```

3. If you make any stanza corrections, reset each monitor checkpoint on the forwarder.

```
cd ~/splunkforwarder/bin

./splunk stop

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
    --file /opt/log/vmail/iis_vmail2.log --reset

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
    --file /opt/log/vmail/iis_vmail3.log --reset

./splunk start
```

NOTE: The **`btprobe`** commands should each be typed all on one line.

4. If you still don't get results, ask your instructor for help.

Module 12 Lab – Create a New Source Type

Description

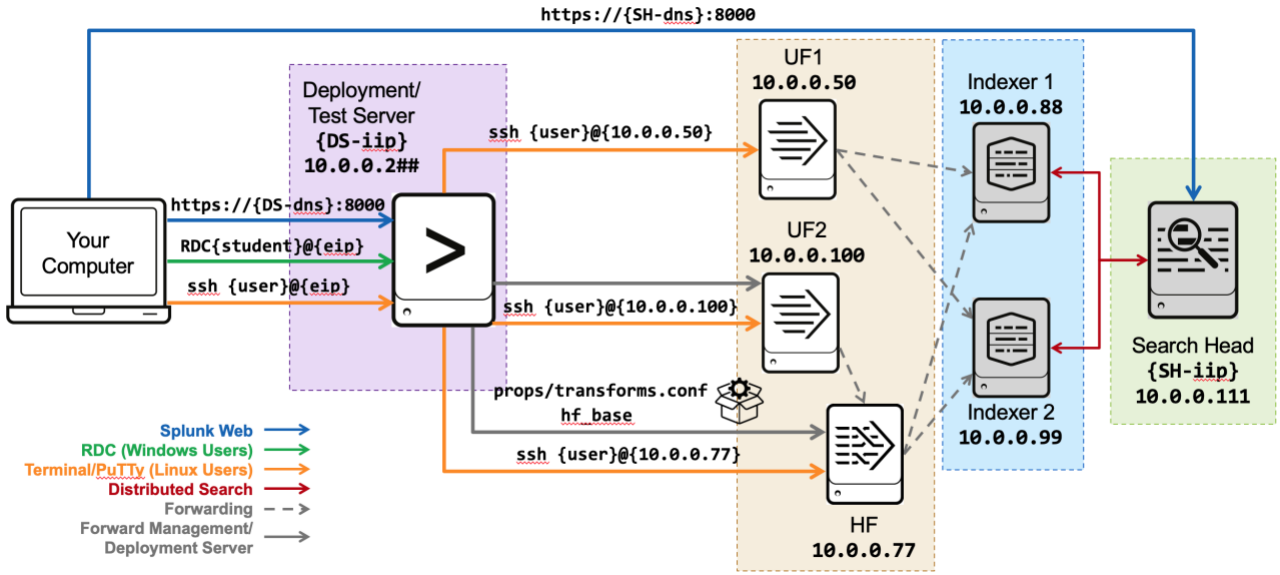
In this exercise, you create two custom source types from two types of data files. The files on the UF2 are considered the production logs. In the lab environment, the deployment server contains the same log files as the forwarders. In a real-world environment, you would need to obtain samples of a production server's data files and manually copy them to the deployment server's or other testing server's file system if you wanted to use the Data Preview feature.

Normally, using a dedicated deployment server, the provisioning steps are:

- On the deployment server, you configure the parsing attributes in **`props.conf`** to process a custom sourcetype using the data preview.
- On the deployment server, you add the same custom sourcetype as a selectable sourcetype.
- Using an appropriate distribution mechanism, you deploy the **`props.conf`** file generated by the Data Preview feature to your indexers. The distribution mechanism depends upon whether your indexers are clustered or non-clustered.

Each forwarder sends its event data marked with the sourcetype to the indexers. During parsing, the indexers extract the proper timestamps and set event boundaries according to the props.conf stanza configurations.

During this lab exercise, you will configure a heavy forwarder (**`10.0.0.77`**) as an intermediate forwarder to receive data from UF2 and parse the data before it is forwarded to the indexers. Therefore, you create and maintain the **`props.conf`** file on the deployment server and deploy it to the heavy forwarder. If you were sending data directly from UF2 to the Indexers, then the **`props.conf`** entries and sourcetype definitions would be on the Indexers, since parsing would be performed on those instances.



NOTE: This lab exercise has several tasks and steps. Successful completion is crucial to complete the subsequent lab exercises.

Steps

Task 1: Add a local monitor input on the deployment server.

In this task, you use the **Add Data** wizard's data preview feature to create a local data input and a new source type that contains custom parsing phase attributes. The custom attributes are needed to correctly parse events from a proprietary (not industry standard) log file.

1. From the deployment server's command line, view the existing crash log files.

```
ls /opt/log/crashlog/
```

```
dir C:\opt\log\crashlog\
```

You should see a number of crash logs with the format **`crash-YYYY-MM-DD-HH_MM_SS.log`**, such as **`crash-2022-07-21-00_43_25.log`**. There is also one file named **`dreamcrusher.xml`**.

2. Select and view one of the crash log files to examine the structure of the data:

```
more /opt/log/crashlog/crash-[DATE].log
```

```
more C:\opt\log\crashlog\crash-[DATE].log
```

The entire file represents one event record. The first line contains a header string with some numbers in the format **`[######]`** followed by the timestamp in the format of **`YYYY-MM-DD HH:MM:SS`**.

```
[167154] 2022-07-21 00:46:26
Received fatal signal 6 (Abort
 Cause:
  Signal sent by PID 6241 running under UID 5898.
 Crashing thread: Main Thread
 Registers
   RDI:  [0x00000B0500000C09]
...
 OS: Linux
 Arch: x86-64

 Backtrace:
  [0x04050A000000D000] gsignal + 53 (/lib64/libc.so.6)
  [0x0600000000000000] abort + 373 (/lib64/libc.so.6)
  [0x000C000000000000] ? (/lib64/libc.so.6)
  [0x80000000090300B0] __assert_perror_fail + 0 (/lib64/libc.so.6)
  [0x0F0000000E00B000] _ZN11XmlDocument8addChildERK7XmlNode + 61 (dcrusherd)
  [0x0800000070500C00] _Z18getSearchConfigXMLR11XmlDocumentPKPKc + 544 (dcrusherd)
  [0x0000100000000000] _Z22do_search_process_impliPKPKcP12BundlesSetupb + 6141 (dcrush
erd)
 Linux / usr13.eng.buttercupgames.com / 2.6.32-279.5.2.el6.x86_64 / #1 SMP Fri Aug 24
01:07:11 UTC 2018 / x86_64
 /etc/redhat-release: CentOS release 6.3 (Final)
 glibc version: 2.12
 glibc release: stable
Last errno: 2
Threads running: 1
argv: [dcrusherd --id=dlz9rtmtkpciptotcz9f --ttl=120 --maxout=500000 --maxtime=8640000
 --reduce_freq=10 --user=system-user]
        0: 0A000000 00008A00 60000000 00E000C8
...
        B: 00A00703 00000420 8000E000 0300C600
terminating...
```

Event timestamp

Event record

3. In Splunk Web on your deployment server, click **Settings > Add Data > Monitor**.
4. On the **Select Source** step, click **Files & Directories**.
5. Click **Browse** to navigate and select one of the crash log files (do not select file **dreamcrusher.xml**) and click **Select**:

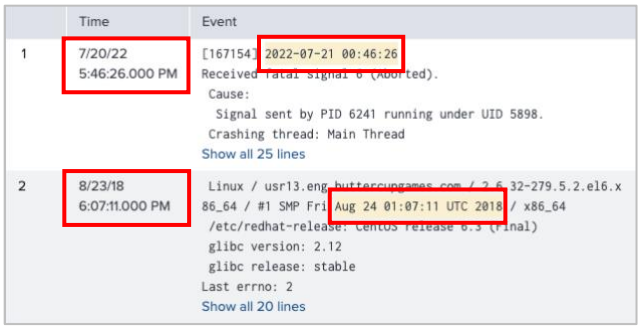
   `/opt/log/crashlog/crash-[DATE].log`

   `C:\opt\log\crashlog\crash-[DATE].log`

6. Verify **Continuously Monitor** is selected and click **Next**.

   On the **Set Source Type** step, note that the ***data preview*** panel displays two events.

7. View the resulting events in the preview, and the timestamps used.

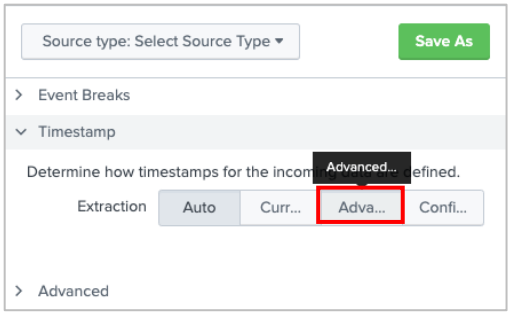


   The crash log consists of information that should be processed as a single event, however you notice that Splunk processes it as two events in the preview.

   The first event is using the timestamp desired for the crashlog, which is listed on the first line.

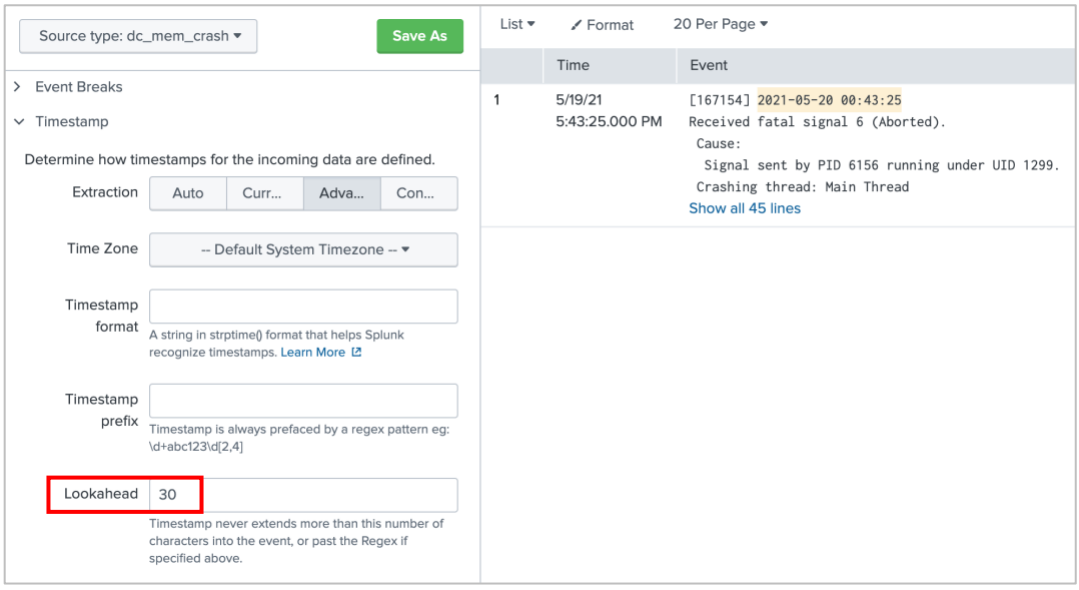
   The second event is created when Splunk finds a timestamp on a later line of the crashlog.

8. To have Splunk treat this as a single event using only the timestamp on the first line, click **Timestamp** > **Advanced…**.



9. Change the **Lookahead** value to **`30`** and press **Tab**.

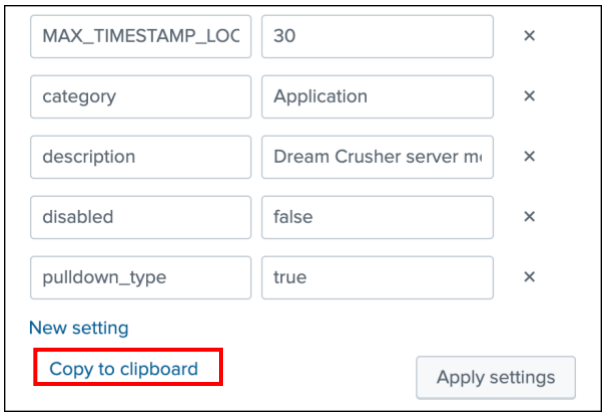
   After the adjustment, the data preview panel should now display only one event.



10. Still on the same step, click **Save As** to save the sourcetype as follows:

    Name: **`dc_mem_crash`**
    Description: **`Dream Crusher server memory dump`**
    Category: **`Application`**
    App: **`Search & Reporting`**

11. Click **Save**.
12. Expand the **Advanced** section on the left and click **Copy to clipboard**.



13. Review the **`props.conf`** attributes produced by your customizations, then click **Cancel**.

    These **`props.conf`** file entries are reviewed again in Task 3.

14. Click **Next** to proceed to **Input Settings**.
15. On the **Input Settings** step, make sure **App Context** is set to **Search & Reporting (search)** and select **`test`** for the Index.

16. Click **Review** and verify that your input matches the following before clicking **Submit**.

Input Type	**File Monitor**
Source Path	**/opt/log/crashlog/crash-XXXX-XX-XX-XX_XX_XX.log**
Continously Monitor	**Yes**
Source Type	**dc_mem_crash**
App Context	**search**
Host	**splunk##**
Index	**test**

17. Click **Start Searching**.

    You should have a single event displayed. If you do, continue to the next task. If not, consult the **Troubleshooting Suggestions** and repeat the task. (You may have to wait several minutes for the result to be displayed)

Task 2: Build an input to index an XML file.

In this task, you create a new data input to parse an XML file. Splunk cannot parse the XML data correctly using the automatic (default) parsing attributes. Use the **Add Data** wizard to create another new custom source type that correctly breaks the XML data into events and extracts a timestamp from within each event.

18. From the deployment server's command line, open the following file in a text editor to examine the structure of the XML data:

```
/opt/log/crashlog/dreamcrusher.xml
```

```
C:\opt\log\crashlog\dreamcrusher.xml
```

    Each **<Interceptor>** node represents a legitimate event record.

    The **<ActionDate>** tag contains the event timestamp in EST time zone.

```
<?xml version="1.0" encoding="UTF-8" ?>
<dataroot>
  <Interceptor>
        <AttackCoords>-80.33100097073213,25.10742916222947</AttackCoords>
        <Outcome>Interdiction</Outcome>
        <Infiltrators>23</Infiltrators>
        <Enforcer>Ironwood</Enforcer>
        <ActionDate>2022-07-07</ActionDate>
        <RecordNotes></RecordNotes>
        <NumEscaped>0</NumEscaped>
        <LaunchCoords>-80.23429525620114,24.08680387475695</LaunchCoords>
        <AttackVessel>Rustic</AttackVessel>
  </Interceptor>
  <Interceptor>
        <AttackCoords>-80.14622349209523,24.53605142362535</AttackCoords>
```

Event timestamp

Event record

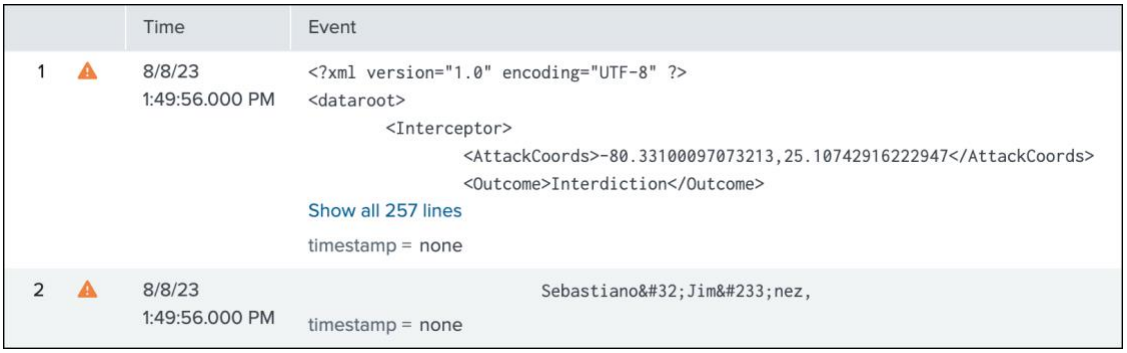
19. Exit the text editor without making any changes to the file.
20. In Splunk Web on your deployment server, launch the **Settings > Add Data** wizard and add a new **Monitor** input.
21. On the **Select Source** step, click **Files & Directories**.
22. Click **Browse** to navigate to the full path to the **dreamcrusher.xml** file and click **Select**.

23. Leave the default for **Continuously Monitor**, and then click **Next**.

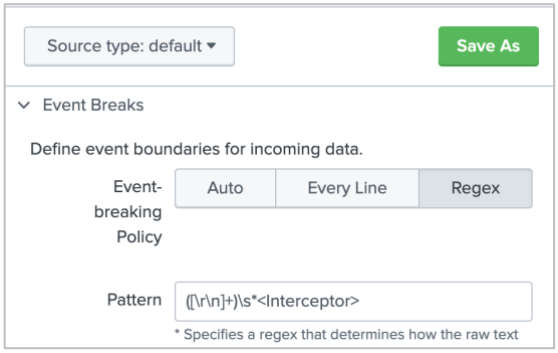
    `/opt/log/crashlog/dreamcrusher.xml`

    `C:\opt\log\crashlog\dreamcrusher.xml`

24. On the **Set Source Type** step, notice the auto event breaking of the XML file is not parsing the file correctly. You'll need to define custom attributes to correct this situation.

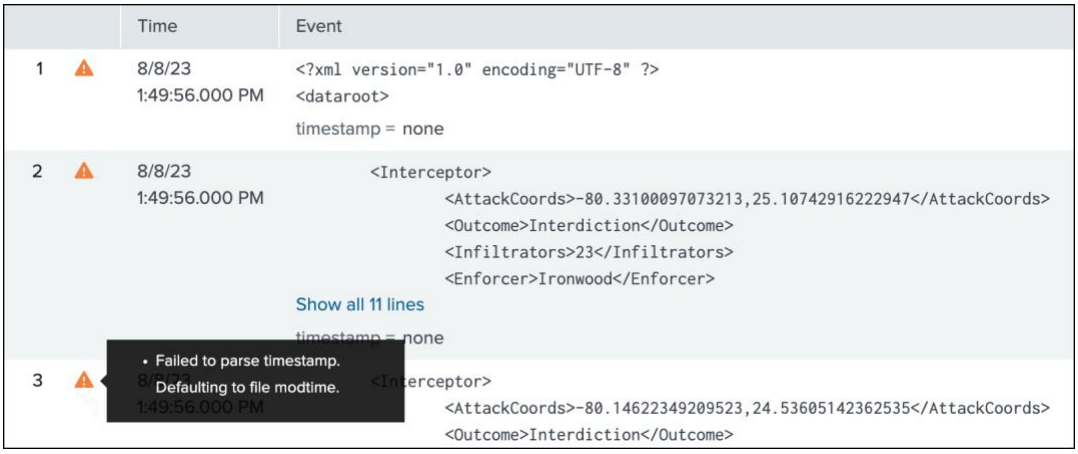


25. Configure the event breaking by expanding the **Event Breaks** section, click **Regex...** and for **Pattern,** type: **`([\r\n]+)\s*<Interceptor>`**.



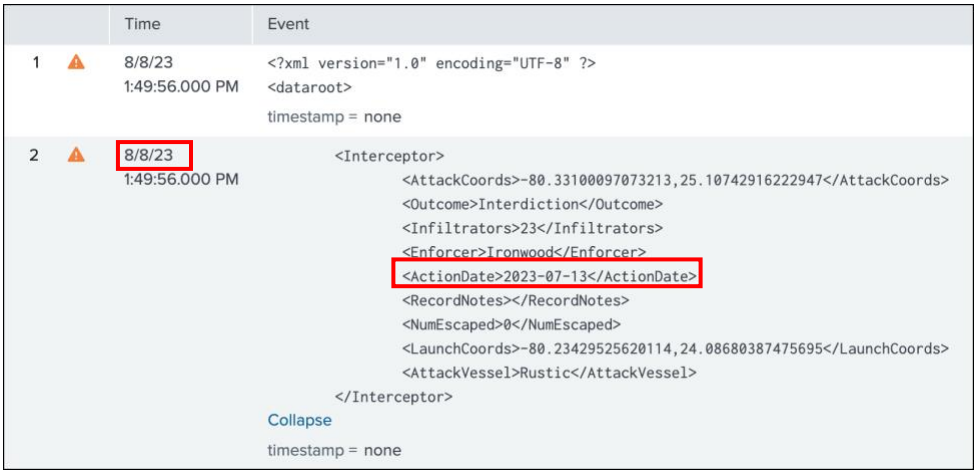
26. Press the **Tab** key to see the effects of the **Event Breaks** settings on the data preview.

    You should see the XML data placed in proper multi-line events, however the timestamps for all these events are the same, and floating over the warning icon shows "Failed to parse timestamp. Defaulting to file modtime".



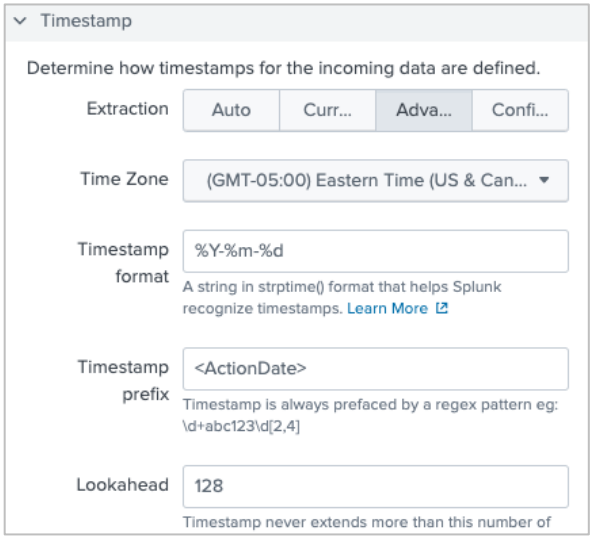
27. To see the **<ActionDate>** tag (timestamp) of the second event, click **Show all ## lines**.

    Notice that the timestamp recorded by Splunk (shown in the **Time** column) does not correctly match the timestamp indicated in the **<ActionDate>** tag in the XML file.



28. Configure the timestamp extraction by expanding the **Timestamp** section and configure as follows.

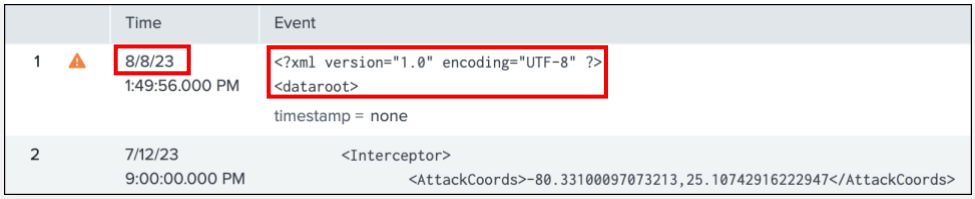
Extraction:	**Advanced…**
Time zone:	**(GMT-5:00) Eastern Time (US & Cananda)**
Timestamp format:	**%Y-%m-%d**
Timestamp prefix:	**<ActionDate>**



29. Press the **Tab** key to see the effect of the **Timestamp** settings on the data preview.

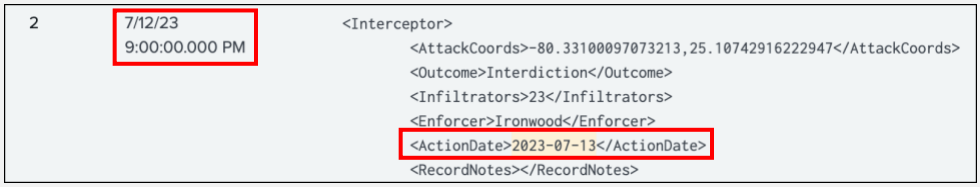
    You should see the dates updated correctly for these events.

NOTE: Using the simple regex **([\r\n]+)\s*<Interceptor>** results in an initial event with XML root elements (starting with **<?xml...>** and **<dataroot>**). Additionally, note that timestamp extraction is not applied to the event. This event can safely be ignored.



Alternatively, you can use a more complex regex to remove these XML root elements and prevent them from being indexed. If desired, use an Event Break Regex Pattern of **`(<.xml.*[\r\n]+)|(<dataroot>)|([\r\n]+)\s<Interceptor>`** to prevent these elements from being indexed.

NOTE: The **ActionDate** time and the **Time** column may not match exactly, since the time is converted from Eastern Standard Time to your local time zone as set on this Splunk instance.



30. Click **Save As** (you may need to scroll to the top of the left column to see the **Save As** button) and save the source type configuration as follows:

Name:	**`dcrusher_attacks`**
Description:	**`Dream Crusher user interactions`**
Category:	**`Application`**
App:	**`Search & Reporting`**

31. Click **Save**, then **Next**.
32. On the **Input Settings** step, make sure the **App Context** is set to **Search & Reporting (search)** and select the **`test`** index then click **Review**.
33. Verify the **Review** page matches the following:

Input Type	**`File Monitor`**	
Source Path	**`/opt/log/crashlog/dreamcrusher.xml`**	(Linux)
	`C:\opt\log\crashlog\dreamcrusher.xml`	(Windows)
Continuously Monitor	**`Yes`**	
Source Type	**`dcrusher_attacks`**	
App Context	**`search`**	
Host	**`splunk##`**	
Index	**`test`**	

34. Click **Submit**.
35. Click **Start Searching**.

    Ignore the XML header event containing **`<?xml version...`**

    If every other event starts with **`<Interceptor>`**, displays the correct timestamp, and the sourcetype is set to **`dcrusher_attacks`**, continue with the next task.

If the format does not match correctly, consult the **Troubleshooting Suggestions** and repeat the task.

Task 3: Prepare the props.conf file on the deployment/test server.

36. In the terminal window connected to the deployment server, copy the contents of the **`props.conf`** file to the **`hf_base`** directory.

```
cp /opt/splunk/etc/apps/search/local/props.conf /opt/splunk/etc/deployment-apps/hf_base/local
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
xcopy "C:\Program Files\Splunk\etc\apps\search\local\props.conf" "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

NOTE: The **`cp`** or **`xcopy`** command should each be typed all on one line.

37. Reload the deployment server. (Splunk may ask you to login as the **`admin`** Splunk user).

```
/opt/splunk/bin/splunk reload deploy-server
```

```
"C:\Program Files\Splunk\bin\splunk" reload deploy-server
```

38. From your deployment/test server, connect to HF (**`10.0.0.77`**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

39. Make sure the new **[dc_mem_crash]** and **[dcrusher_attacks]** stanzas appear in the deployed **props.conf** file.

    Your **dc_mem_crash** and **dcrusher_attacks** stanzas should match the output shown below:

```
cat ~/splunk/etc/apps/hf_base/local/props.conf
...

[dc_mem_crash]
DATETIME_CONFIG =
LINE_BREAKER = ([\r\n]+)
MAX_TIMESTAMP_LOOKAHEAD = 30
NO_BINARY_CHECK = true
category = Application
description = Dream Crusher server memory dump
pulldown_type = true

[dcrusher_attacks]
BREAK_ONLY_BEFORE_DATE =
DATETIME_CONFIG =
LINE_BREAKER = ([\r\n]+)\s*<Interceptor>
NO_BINARY_CHECK = true
SHOULD_LINEMERGE = false
TIME_FORMAT = %Y-%m-%d
TIME_PREFIX = <ActionDate>
TZ = America/New_York
category = Application
description = Dream Crusher user interactions
disabled = false
pulldown_type = true
```

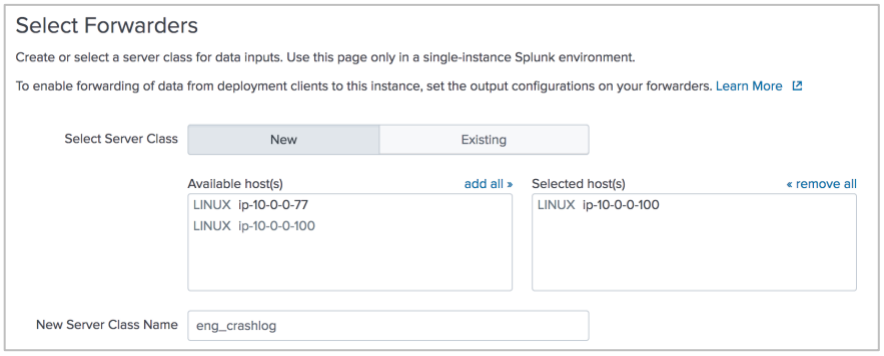
40. Exit the HF using the **exit** command.

Task 4: Deploy a directory monitor to UF2 to index the crash logs into the test index.

In this task, you will create a remote input with the **Add Data** wizard to monitor the crashlog files from UF2. You will need to exclude the **dreamcrusher.xml** file while creating the remote input.

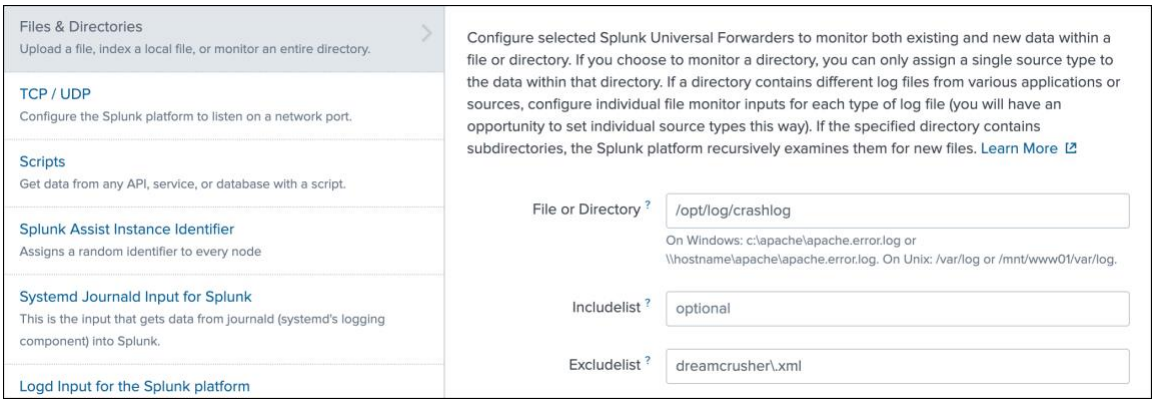
41. In Splunk Web on your deployment server, click **Settings > Add Data > Forward**.
42. On the **Select Forwarders** step, configure the form as follows, and click **Next**:

    Select Server Class: **New**
    Selected host(s): **LINUX ip-10-0-0-100**
    New Server Class Name: **eng_crashlog**



43. On the **Select Source** step, click **Files & Directories** click **Files & Directories** and configure the form as follows, and click **Next**:

    File or Directory: **`/opt/log/crashlog`**
    Excludelist: **`dreamcrusher\.xml`**

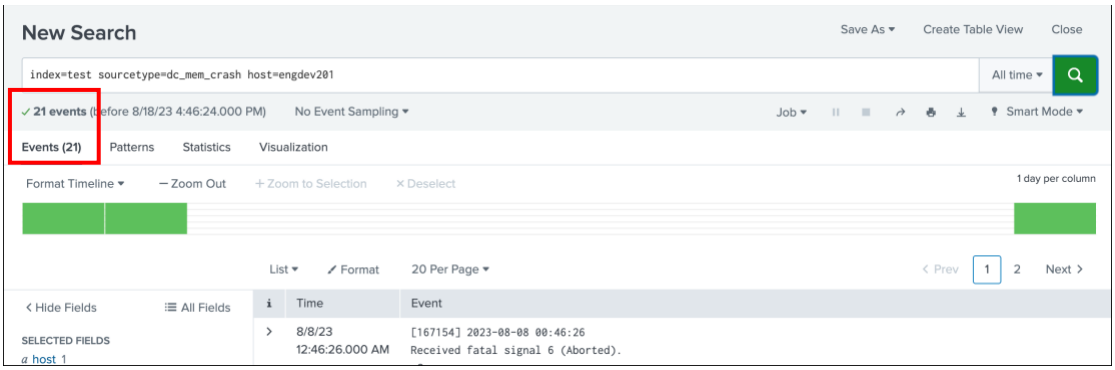


44. For the **Input Settings**, for the Source type click on **Select** and select **Application > `dc_mem_crash`** sourcetype defined earlier, and the **`test`** index, then click **Review**.
45. Verify the **Review** page matches the following, then click **Submit**:

    New Server Class Name **`eng_crashlog`**
    Selected host(s) **`LINUX | ip-10-0-0-100`**
    Input Type **`File Monitor`**
    Source Path **`/opt/log/crashlog`**
    Includelist **`N/A`**
    Excludelist **`dreamcrusher\.xml`**
    Source Type **`dc_mem_crash`**
    Index **`test`**

46. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID):

    `index=test sourcetype=dc_mem_crash host=engdev2##`



The number of entries you see should equal one entry per crash log file in the **`/opt/log/crashlog`** folder on the UF2 instance. To verify, login to UF2 (**`10.0.0.100`**) using a terminal window, and run the command: **`ls /opt/log/crashlog/crash*.log`**. For example, if you see 21 source files, you should see 21 events.

If instead you see multiple events per source file, verify your configurations by consulting the **Troubleshooting Suggestions** and repeat the task.

Task 5: Deploy a file monitor to UF2 to transmit the dreamcrusher.xml data.

In this task, you add a Forward input to monitor **`dreamcrusher.xml`** on UF2. The XML file is forwarded to your heavy forwarder for line breaking and timestamp extraction. The parsed events are then forwarded to the indexers.

47. In Splunk Web on your deployment server, launch the **Settings > Add Data** wizard and add a **Forward** input to monitor **`dreamcrusher.xml`** on UF2 (**`10.0.0.100`**). Send the data to the **`test`** index.

    - On the **Select Forwarders** step:

Selected Server Class	**New**
Selected host(s)	**`LINUX|ip-10-0-0-100`**
New Server Class Name	**`eng_dreamcrusherXML`**

    - On the **Select Source** step, select **Files & Directories**:

File or Directory	**`/opt/log/crashlog/dreamcrusher.xml`**

    - On the **Input Settings** step:

Source type	**Select**, using **Application > `dcrusher_attacks`**
Index	**`test`**

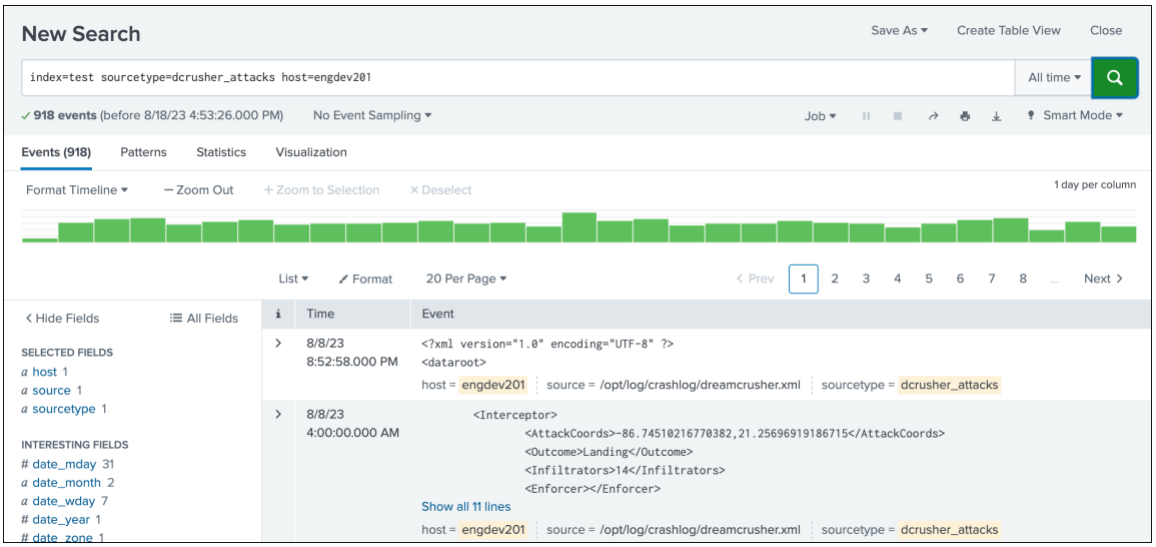
48. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	**`eng_dreamcrusherXML`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log/crashlog/dreamcrusher.xml`**
Includelist	**`N/A`**
Excludelist	**`N/A`**
Source Type	**`dcrusher_attacks`**
Index	**`test`**

49. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID):

    `index=test sourcetype=dcrusher_attacks host=engdev2##`

    Note that it can take a few minutes for the data to display on the search head.



    You should now see a total of 918 events. Except for the first event, all events should begin with the **`<Interceptor>`** tag.

    The total event count for **`dcrusher_attacks`** sourcetype is **918**. If you get a different count, why is that?

    If the event count is not **918**, there could be several reasons:

    - Not searching over **All time**.
    - Misconfigured event breaking.
    - Verify the events from the UF2 (**host=engdev2##**) are the only ones being searched.

Troubleshooting Suggestions

1. For task 1, verify the **props.conf** file located in the **SPLUNK_HOME/etc/apps/search/local** directory on the deployment/test server has the following stanza:

```
[dc_mem_crash]
DATETIME_CONFIG =
LINE_BREAKER = ([\r\n]+)
MAX_TIMESTAMP_LOOKAHEAD = 30
NO_BINARY_CHECK = true
category = Application
description = Dream Crusher server memory dump
pulldown_type = true
```

2. For task 2, verify the **props.conf** file located in the **SPLUNK_HOME/etc/apps/search/local** directory on the deployment/test server has the following stanza for **dcrusher_attacks** in addition to the **dc_mem_crash** stanza:

```
[dcrusher_attacks]
BREAK_ONLY_BEFORE_DATE =
DATETIME_CONFIG =
LINE_BREAKER = ([\r\n]+)\s*<Interceptor>
NO_BINARY_CHECK = true
SHOULD_LINEMERGE = false
TIME_FORMAT = %Y-%m-%d
TIME_PREFIX = <ActionDate>
TZ = America/New_York
category = Application
description = Dream Crusher user interactions
disabled = false
pulldown_type = true
```

NOTE: After you deploy the **props.conf** to the heavy forwarder, **props.conf** file should have the **dc_mem_crash** and **dcrusher_attacks** stanzas.

3. If you are not seeing data, verify **inputs.conf** for **crashlog** and **dreamcrusher.xml** on your deployment/test server,

```
more /opt/splunk/etc/deployment-apps/_server_app_eng_crashlog/local/inputs.conf
```

```
[monitor:///opt/log/crashlog]
blacklist = dreamcrusher\.xml
disabled = false
index = test
sourcetype = dc_mem_crash
```

```
more /opt/splunk/etc/deployment-apps/_server_app_eng_dreamcrusherXML/local/inputs.conf
```

```
[monitor:///opt/log/crashlog/dreamcrusher.xml]
disabled = false
index = test
sourcetype = dcrusher_attacks
```

4. If you make any stanza corrections, reset the monitor checkpoints on UF2.

```
cd ~/splunkforwarder/bin
./splunk stop
./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/crashlog/dreamcrusher.xml --reset

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/crashlog/crash-<xxxx-xx-xx-xx_xx_xx>.log --reset

./splunk start
```

NOTE: The **btprobe** command is shown across two lines but it should be entered on a single line.

Replace **<**xxxx**...>** with the actual file name.

5. If you still don't get results, ask your instructor for help.

Module 13 Lab – Manipulating Input Data

Description

In this lab exercise, you use both Ingest Actions and legacy methods (modifying **`props.conf`** and **`transforms.conf`**) to mask various sensitive data from two different log files on the Deployment / Test Server, and then verify this behavior when ingesting data from UF2 through the HF intermediate forwarder.

IMPORTANT: You can perform Ingest Action lab steps (tasks 1-3) using either a Linux or Windows Deployment Server, however in Splunk 9.0 in production only Linux is supported with Ingest Actions. For updates on supported platforms with Ingest Actions, refer to: https://docs.splunk.com/Documentation/Splunk/latest/Data/DataIngest.

Steps

Task 1: Create a data masking transformation.

1. On the deployment/test server view the sales entries log file using a text editor or **`more`** command (dates may be different from those shown in the below screen shot).

```
more /opt/log/ecommsv1/sales_entries.log
```

```
more C:\opt\log\ecommsv1\sales_entries.log
```

   Review the event data:

```
Tue Jun 21 2022 16:24:11 Sent to checkout TransactionID=100763
Tue Jun 21 2022 16:24:11 checkout response for TransactionID=100763 CustomerID=6i30kqk3
Tue Jun 21 2022 16:24:11 ecomm engine response TransactionID=100763 CustomerID=6i30kqk3 accepted
Tue Jun 21 2022 16:24:12 TransactionID=100763 AcctCode=8333-4577
Tue Jun 21 2022 16:24:13 Sent to Accounting System 100303
...
```

   Quit the **`more`** command with '**`q`**' or close the file when done.

2. For this exercise, download the sample log file (**`sales_entries_samples.log`**), which has a smaller set of these log entries, from **`https://splk.it/edu-data-admin-91`**.
3. In Splunk Web on the deployment server, navigate to **Settings > Source types**.
4. Click **New Source Type**.
5. On the **Create Source Type** dialog, configure the form as follows (leaving all other fields as their defaults) and then click **Save**:

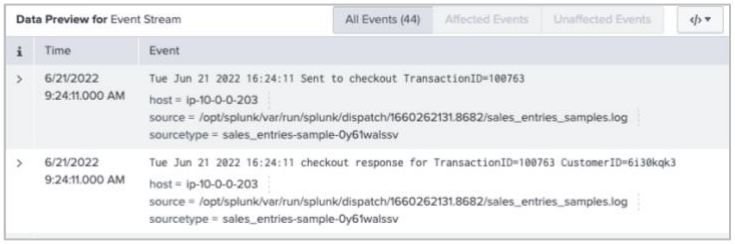
Name:	**`sales_entries`**
Destination app:	**`Search & Reporting`**
Category:	**`Custom`**

6. Navigate to **Settings > Ingest actions**.
7. Click **New Ruleset**.
8. On the **Create New Ruleset** page, configure the form as follows:

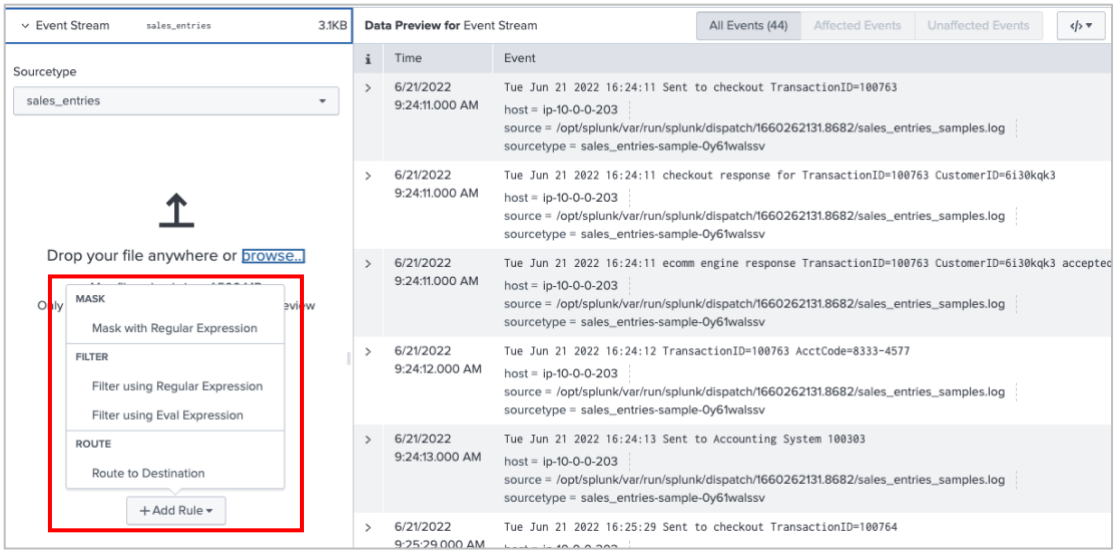
Enter Ruleset Name:	**`Mask Sales Entries`** (*When typing the name, spaces are converted to underscores*)
Enter Ruleset Description:	**`Mask account codes in the sales_entries.log`**

9. Under **Preview using**, select **Sample File**.
10. In the **Sourcetype** drop down, select **Custom > sales_entries**.
11. Under the **Sourcetype** dropdown click on the hyperlink for the word "**browse…**" and then select the lab file you previously downloaded in step 2: **`sales_entries_samples.log`**.

    You see resulting entries under **Data Preview for Event Stream** in the right-hand side of the browser.

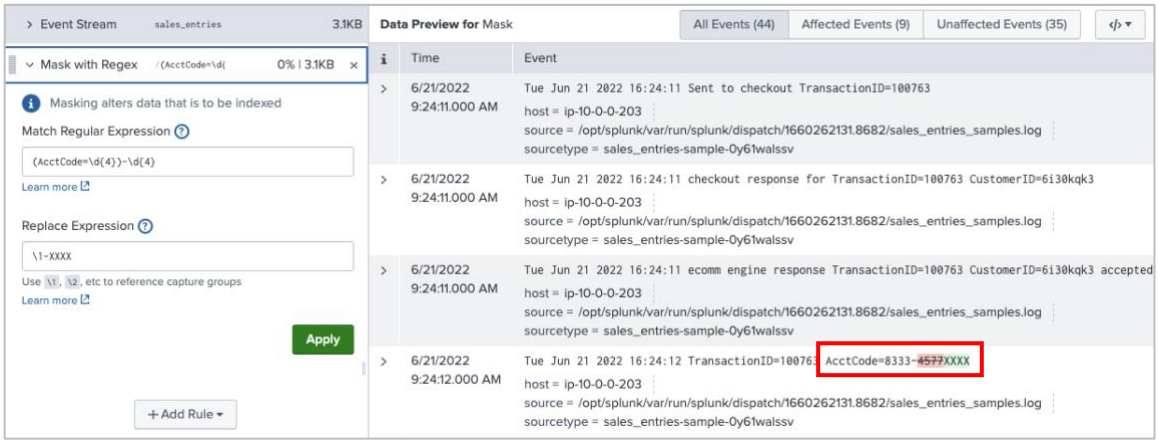


12. Click **Add Rule > Mask with Regular Expression** at the bottom left side of the screen.



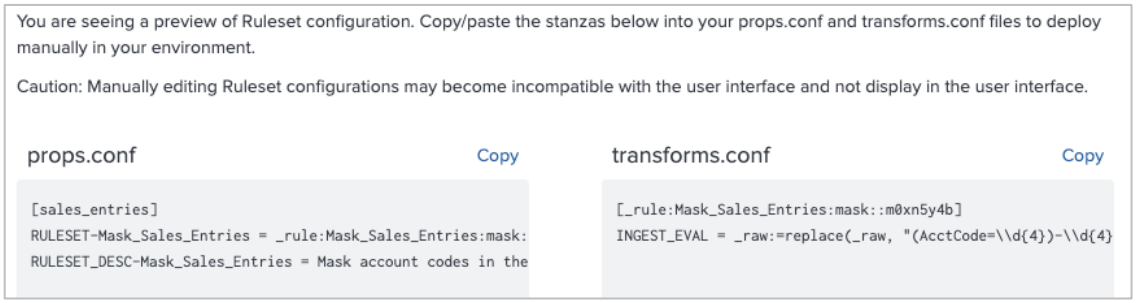
13. Fill in the **Mask with Regex** section with the following, then click **Apply**:

Match Regular Expression	**`(AcctCode=\d{4})-\d{4}`**
Replace Expression	**`\1-XXXX`**



Entries in the **Data Preview** section which contain the **AcctCode** now show the original data that will be masked as highlighted with strikethrough, and the replacement text in green highlight, similar to the following: **AcctCode=8333-~~4577~~XXXX**.

14. Click **Preview Config** in the top right corner and view the resulting **props.conf** and **transforms.conf** entries, then click **Close**.



In the **props.conf** text you see a stanza for the **sales_entries** source type that defines the ruleset name and description. In the **transforms.conf** text you see the rule associated with that ruleset, which shows a Regex replacement.

15. Click **Save** in the top right corner.
16. In the Ingest Actions table, click on the arrow (**>**) under the information (**i**) column to view the description and rules associated with the **Mask_Sales_Entries** ruleset.

Task 2: View deployment-apps files for the splunk_ingest_actions app

In this task you view the **splunk_ingest_actions** deployment app that was created on the deployment server, and how it has been replicated to the managed heavy forwarder (HF). In a production environment you would want to ensure this app is replicated to all Splunk servers involved with parsing the data, including any indexers. In these lab tasks only the heavy forwarder will perform this parsing for our data for Ingest Actions.

17. On the console of the deployment server view the **props.conf** for **splunk_ingest_actions** deployment app.

```
cd /opt/splunk/etc/deployment-apps/
more splunk_ingest_actions/local/props.conf
```

```
cd "C:\Program Files\Splunk\etc\deployment-apps"
more splunk_ingest_actions\local\props.conf
```

You should see entries similar to the following (with a randomized alphanumberic string *xxxxxxxx*), which match the **Preview Config** dialog you saw in the last task:

```
[sales_entries]
RULESET-Mask_Sales_Entries = _rule:Mask_Sales_Entries:mask::xxxxxxxx
RULESET_DESC-Mask_Sales_Entries = Mask account codes in the sales_entries.log
```

18. View the **transforms.conf** for **splunk_ingest_actions** deployment app.

```
more splunk_ingest_actions/local/transforms.conf
```

```
more splunk_ingest_actions\local\transforms.conf
```

You should an entry similar to the following:

```
[_rule:Mask_Sales_Entries:mask::xxxxxxxx]
INGEST_EVAL = _raw:=replace(_raw, "(AcctCode=\\d{4})-\\d{4}", "\\1-XXXX")
```

Quit the **more** command with '**q**' or close the file when done.

19. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

20. On the HF view the **props.conf** for **splunk_ingest_actions** deployment app.

```
cd ~/splunk/etc/apps/
cat splunk_ingest_actions/local/props.conf
```

You should see the same entries as was listed on the deployment server:

```
[sales_entries]
RULESET-Mask_Sales_Entries = _rule:Mask_Sales_Entries:mask::xxxxxxxx
RULESET_DESC-Mask_Sales_Entries = Mask account codes in the sales_entries.log
```

21. On the HF view the **transforms.conf** for **splunk_ingest_actions** deployment app.

```
cat splunk_ingest_actions/local/transforms.conf
```

You should see the same entries as was listed on the deployment server:

```
[_rule:Mask_Sales_Entries:mask::xxxxxxxx]
INGEST_EVAL = _raw:=replace(_raw, "(AcctCode=\\d{4})-\\d{4}", "\\1-XXXX")
```

22. Exit the HF using the **exit** command.

Task 3: Create data input on UF2 for sales_entries.log

23. In Splunk Web on your deployment server, launch the **Settings > Add Data** wizard and add a **Forward** input to monitor **`sales_entries.log`** on UF2 (**`10.0.0.100`**). Send the data to the **`sales`** index.

    - On the **Select Forwarders** step:

Selected Server Class	**New**
Selected host(s)	**`LINUX|ip-10-0-0-100`**
New Server Class Name	**`eng_sales_entries`**

    - On the **Select Source** step, select **Files & Directories**:

File or Directory	**`/opt/log/ecommsv1/sales_entries.log`**

    - On the **Input Settings** step:

Source type	**Select**, using **Custom > `sales_entries`**
Index	**`sales`**

24. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	**`eng_sales_entries`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log/ecommsv1/sales_entries.log`**
Includelist	**`N/A`**
Excludelist	**`N/A`**
Source Type	**`sales_entries`**
Index	**`sales`**

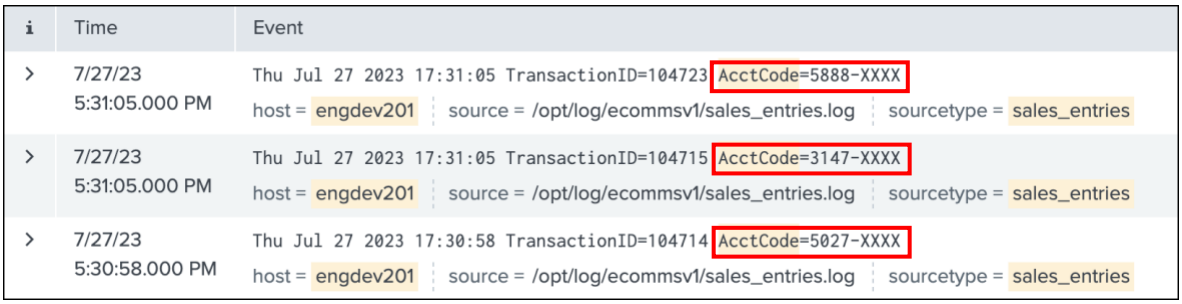
25. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID) to verify data has been ingested:

    `index=sales sourcetype=sales_entries host=engdev2##`

NOTE: It can take a few minutes for the data to display on the search head.

26. Add the text **`AcctCode`** to the search and confirm masked entries configured with Ingest Actions:

    `index=sales sourcetype=sales_entries host=engdev2## AcctCode`



All instances of events with **`AcctCode`** should have the last 4 digits masked as **`XXXX`**.

Task 4: Create a data masking transformation manually with transforms.conf and props.conf

In this task you will manually create entries in **transforms.conf** and **props.conf** to perform data masking. In these tasks you will create these entries on the deployment server.

27. In a terminal window for the deployment/test server, open **vendor_sales.log** file in a text editor.

```
more /opt/log/vendorUS1/vendor_sales.log
```

```
more C:\opt\log\vendorUS1\vendor_sales.log
```

Play close attention to the **AcctID** field:

```
[15/Aug/2022:22:08:44] VendorID=1082 Code=J AcctID=9626328233280520
[15/Aug/2022:22:18:31] VendorID=1103 Code=N AcctID=1236221558889916
...
```

Quit the **more** command with '**q**' or close the file when done.

NOTE: The **AcctID** field values contain sensitive account numbers; these numbers should be masked for privacy reasons.

28. Create a **transforms.conf** file using a text editor:

```
/opt/splunk/etc/apps/search/local/transforms.conf
```

```
C:\Program Files\Splunk\etc\apps\search\local\transforms.conf
```

29. Add the following stanza to mask the last six digits of the 16-digit **AcctID** field values:

```
[mask-acctid]
REGEX = (.*AcctID=\d{10})\d{6}
DEST_KEY = _raw
FORMAT = $1XXXXXX
```

30. Open the following **props.conf** file using a text editor:

```
/opt/splunk/etc/apps/search/local/props.conf
```

```
C:\Program Files\Splunk\etc\apps\search\local\props.conf
```

31. Append the following stanza to invoke the **acctmasking** transformations for the **vendor_sales_entries** source type:

```
[vendor_sales]
TRANSFORMS-acctmasking = mask-acctid
```

32. Restart the deployment server.

```
/opt/splunk/bin/splunk restart
```

```
"C:\Program Files\Splunk\bin\splunk" restart
```

Task 5: Add a local file monitor input to test the transformation.

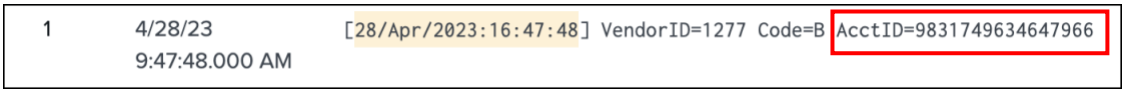
33. Log back into the deployment/test server and launch the **Settings > Add Data** wizard.
34. Add a **Monitor** (local) input and, on the **Select Source** step, select **Files & Directories**.
35. Click **Browse** and select the following file:

```
/opt/log/vendorUS1/vendor_sales.log
```

```
C:\opt\log\vendorUS1\vendor_sales.log
```

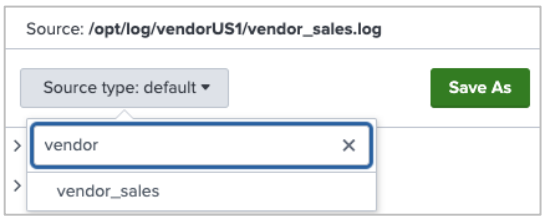
36. Verify **Continuously Monitor** is selected and click **Next**.
37. Look for an event entry with an **AcctID** in the data preview pane.

    Notice that the **AcctID** is currently *not* masked.



38. Click **Source type: default** and type "**vendor**".

    The **vendor_sales** source type should display.



39. Select **vendor_sales** for the source type.

    After selecting the **vendor_sales** sourcetype, the **AcctID** values in the data preview pane should be masked.

IMPORTANT: If the **`AcctID`** values are not masked, then QUIT the **Add Data** wizard by clicking on the Splunk logo. Verify the syntax and spelling carefully in **`transforms.conf`** and **`props.conf`** (see Task 4), restart Splunk and repeat this step.

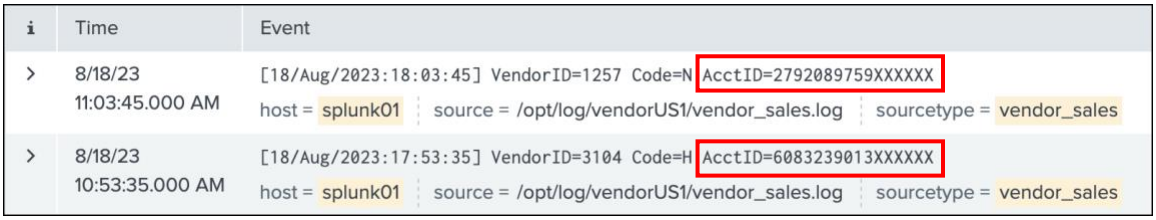
40. Click **Next** and select **Search & Reporting (search)** for the **App Context**, and **`test`** for the **Index** setting.
41. Click **Review** and verify that your input matches the following before clicking **Submit**.

Input Type	**`File Monitor`**
Source Path	**`/opt/log/vendorUS1/vendor_sales.log`** **`C:\opt\log\vendorUS1\vendor_sales.log`**
Continously Monitor	**`Yes`**
Source Type	**`vendor_sales`**
App Context	**`search`**
Host	**`splunk##`**
Index	**`test`**

42. From the deployment/test server, execute the following search over the **Last 7 days** (replacing the ## with your student ID):

    ```
    index=test sourcetype=vendor_sales host=splunk##
    ```

    You should see events with the last six digits of the **AcctID** field masked.



Task 6: Copy the props and transforms file definitions to the hf_base app and deploy them to the HF.

43. On the deployment server, copy the contents of the **props.conf** file to the **hf_base** directory.

```
cp /opt/splunk/etc/apps/search/local/props.conf /opt/splunk/etc/deployment-apps/hf_base/local/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
copy "C:\Program Files\Splunk\etc\apps\search\local\props.conf" "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

NOTE: The **cp** or **xcopy** command should each be typed all on one line.

44. On the deployment server, copy the contents of the **transforms.conf** file to the **hf_base** directory.

```
cp /opt/splunk/etc/apps/search/local/transforms.conf /opt/splunk/etc/deployment-apps/hf_base/local/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
copy "C:\Program Files\Splunk\etc\apps\search\local\transforms.conf" "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

NOTE: The **cp** or **xcopy** command should each be typed all on one line.

45. Reload the deployment server.

    Splunk may ask you to login as the **admin** Splunk user.

```
/opt/splunk/bin/splunk reload deploy-server
```

```
“C:\Program Files\Splunk\bin\splunk” reload deploy-server
```

46. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

47. Confirm the new **[sales_entries]** stanza below appears with the other stanzas in the deployed **props.conf** file. (It may take up to 30 seconds before you see the change.)

```
cat ~/splunk/etc/apps/hf_base/local/props.conf

[dc_mem_crash]
DATETIME_CONFIG =
LINE_BREAKER = ([\r\n]+)
MAX_TIMESTAMP_LOOKAHEAD = 30
NO_BINARY_CHECK = true
...


[vendor_sales]
TRANSFORMS-acctmasking = mask-acctid
```

48. Confirm the new stanza appears in the deployed **transforms.conf** file.

```
cat ~/splunk/etc/apps/hf_base/local/transforms.conf

[mask-acctid]
REGEX = (.*AcctID=\d{10})\d{6}
DEST_KEY = _raw
FORMAT = $1XXXXXX
```

49. Exit the HF.

```
exit
```

Task 7: Deploy a file monitor to UF2 to transmit the vendor_sales.log data.

50. Launch the **Settings > Add Data** wizard on the deployment/test server and add a **Forward** input to monitor **vendor_sales.log** on UF2. Send the data to the **itops** index.

- On the **Select Forwarders** step:
  Selected Server Class **New**
  Selected host(s) **LINUX|ip-10-0-0-100**
  New Server Class Name **eng_vendorsales**
- On the **Select Source** step, select **Files & Directories**:
  File or Directory **/opt/log/vendorUS1/vendor_sales.log**
- On the **Input Settings** step:
  Source type **Select**, type "**vendor**" and select **vendor_sales**
  Index **sales**

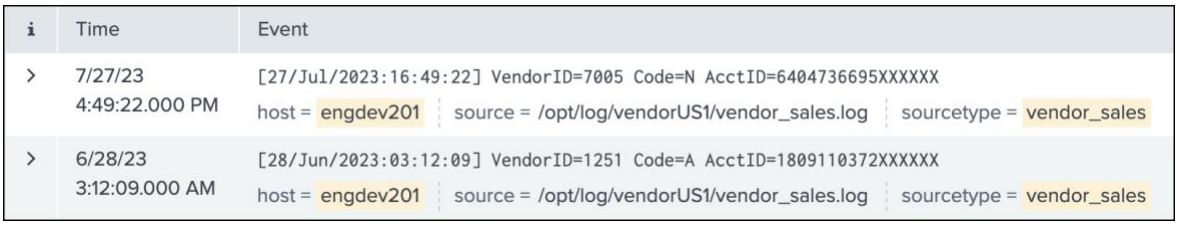
51. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	**eng_vendorsales**
List of Forwarders	**LINUX|ip-10-0-0-100**
Input Type	**File Monitor**
Source Path	**/opt/log/vendorUS1/vendor_sales.log**
Includelist	**N/A**
Excludelist	**N/A**
Source Type	**vendor_sales**
Index	**sales**

52. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID):

```
index=sales sourcetype=vendor_sales host=engdev2##
```

You should see events from UF2 with the last six digits of the **AcctCode** field masked.



NOTE: It may take a few minutes for the results to appear.

Troubleshooting Suggestions

If your searches are not producing the expected results, check your configurations.

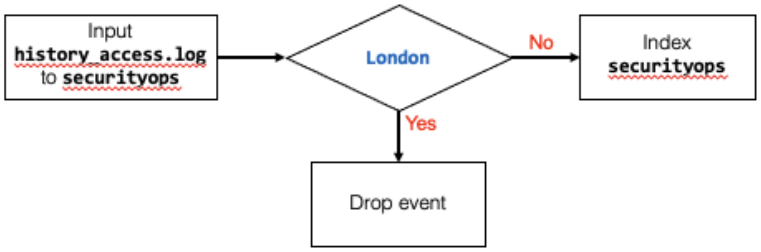
1. Verify the syntax and spelling in all configurations and searches.

Module 14 Lab – Routing Input Data

Description

In this lab exercise, you use both Ingest Actions and legacy methods (modifying **`props.conf`** and **`transforms.conf`**) to route and drop various data from two different log files on the Deployment / Test Server, and then verify this behavior when ingesting data from UF2 through the HF intermediate forwarder.

In the first set of tasks you will use Ingest Actions to create rules to check the badge access logs and drop all events from location **`London`**, and keep all other events in the **`securityops`** index, based on REGEX pattern matches as depicted here:



IMPORTANT: You can perform Ingest Action lab steps (tasks 1-3) using either a Linux or Windows Deployment Server, however in Splunk 9.0 in production only Linux is supported with Ingest Actions. For updates on supported platforms with Ingest Actions, refer to: https://docs.splunk.com/Documentation/Splunk/latest/Data/DataIngest.

Steps

Task 1: Create a data routing ingest action.

In this task, you create an ingest action to take the following actions:

- If an event does not contain the term **`London`**, index the event in the index defined in the data input: **`securityops`**.
- If an event contains the term **`London`**, drop the event (do not index it).

1. On the deployment/test server view the badge access history log file using a text editor or **`more`** command.

```
more /opt/log/badgesv1/history_access.log
```

```
more C:\opt\log\badgesv1\history_access.log
```

Review the event data. Each event contains one of three **`Address_Description`** locations: **`London`**, **`San Francisco`**, or **`Boston`**:

```
Jun 22 2022 8:23:7
Address=1.1.1.R2
Address_Description=London
Device=Proximity Reader
Event_Description=Access Granted: Door Used
rfid=350526492374
...
```

Quit the **`more`** command with '**`q`**' or close the file when done.

2. For this exercise, download the sample log file (**history_access_samples.log**), which has a smaller set of these log entries, from **https://splk.it/edu-data-admin-91**.
3. In Splunk Web on the deployment server, navigate to **Settings > Source types**.
4. Click **New Source Type**.
5. On the **Create Source Type** dialog, configure the form as follows (leaving all other fields as their defaults) and then click **Save**:

   Name: **badge_access**
   Destination app: **Search & Reporting**
   Category: **Custom**
   Indexed extractions: **none**
   Event Breaks
   Event-breaking Policy: **Regex**
   Pattern: **rfid=.*([\r\n]+)**

   These settings will break events after the line with the **rfid** listed. Without adding these event break parameters, Splunk may interpret the **rfid** number as a timestamp and a separate event.

6. Navigate to **Settings > Ingest actions**.
7. Click **New Ruleset**.
8. On the **Create New Ruleset** page, configure the form as follows:

   Enter Ruleset Name: **Badge Access US**
   (*When typing the name, spaces are converted to underscores*)
   Enter Ruleset Description: **Badge access in US cities**

9. In **Preview using**, select **Sample File**.
10. In the **Sourcetype** drop down select **Custom > badge_access**.
11. Under the **Sourcetype** dropdown click on the hyperlink for the word "**browse…**" and then select the lab file you previously downloaded in step 2: **history_access_samples.log** and click **Open**.

    You should see resulting entries under **Data Preview for Event Stream** in the right hand side of the browser.

12. Click **+ Add Rule > Filter using Regular Expression** at the bottom left side of the screen.

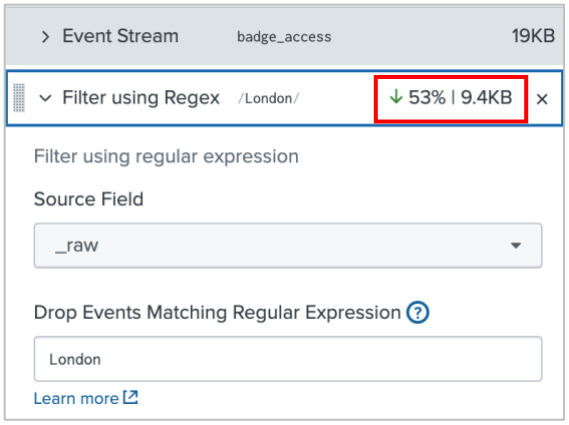
13. Fill in the **Filter using Regular Expression** section with the following:

Source Field	**_raw**
Drop Events Matching Regular Expression	**London**

14. Click **Apply** to view the resulting Data Preview.

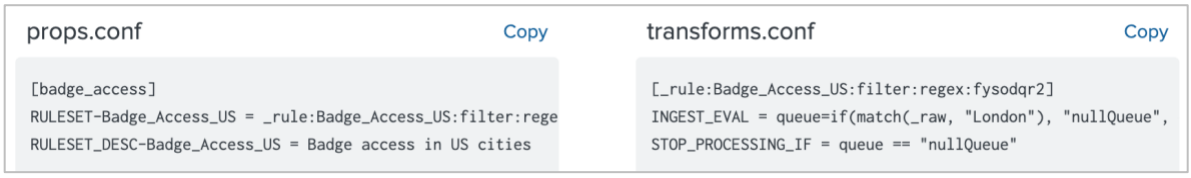
    You should see in the Data Preview section where matching entries to be filtered are highlighted in red.

    Additionally, you should see a reduction in events total ingest size for the data in the left-hand column.



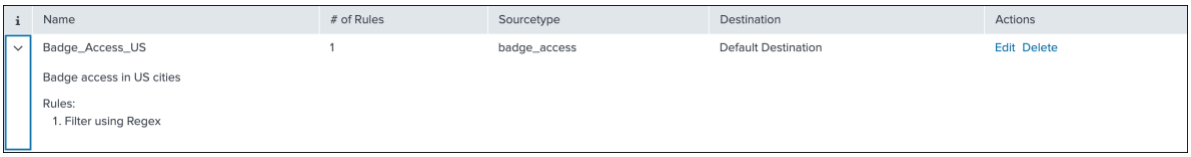
15. To the right of **Data Preview for Filter**, click the **Affected Events** and **Unaffected Events** filter buttons to verify that only events with the keyword **London** are affected (**Boston** and **San Francisco** events should be unaffected).

16. Click **Preview Config** in the top right corner and view the resulting **`props.conf`** and **`transforms.conf`** entries, then click **Close**.



In the **`props.conf`** text you see a stanza for the **`badge_access`** source type that defines the ruleset name and description. In the **`transforms.conf`** text you see the rule associated with that ruleset, which shows an **`INGEST_EVAL`** rule, and the use of the **`nullQueue`**.

17. Click **Save** in the top right corner.
18. In the Ingest Actions table, click the arrow (**`>`**) under the information (**`i`**) column to view the description and rules associated with the **Badge_Access_US** ruleset.



Task 2: View deployment-apps files for the splunk_ingest_actions app

In this task you view the **`splunk_ingest_actions`** deployment app that was created on the deployment server, and how it has been replicated to the managed heavy forwarder (HF). In a production environment you would want to ensure this app is replicated to all Splunk servers involved with parsing the data, including any indexers. In these lab tasks only the heavy forwarder will perform this parsing for our data for Ingest Actions.

19. On the console of the deployment server view the **`props.conf`** for **`splunk_ingest_actions`** deployment app.

```
cd /opt/splunk/etc/deployment-apps/
more splunk_ingest_actions/local/props.conf
```

```
cd "C:\Program Files\Splunk\etc\deployment-apps"
more splunk_ingest_actions\local\props.conf
```

You should see entries similar to the following (with a randomized alphanumberic string *xxxxxxxx*), in addition to any additional entries from prior labs:

```
[badge_access]
RULESET-Badge_Access_US = _rule:Badge_Access_US:filter:regex:xxxxxxxx
RULESET_DESC-Badge_Access_US = Badge access in US cities
```

20. View the **transforms.conf** for **splunk_ingest_actions** deployment app.

```
more splunk_ingest_actions/local/transforms.conf
```

```
more splunk_ingest_actions\local\transforms.conf
```

You should an entry similar to the following:

```
[_rule:Badge_Access_US:filter:regex:xxxxxxxx]
INGEST_EVAL = queue=if(match(_raw, "London"), "nullQueue", queue)
STOP_PROCESSING_IF = queue == "nullQueue"
```

21. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

22. On the HF view the **props.conf** for **splunk_ingest_actions** deployment app.

```
cd ~/splunk/etc/apps/
more splunk_ingest_actions/local/props.conf
```

You should see the same entries as was listed on the deployment server:

```
[badge_access]
RULESET-Badge_Access_US = _rule:Badge_Access_US:filter:regex:xxxxxxxx
RULESET_DESC-Badge_Access_US = Badge access in US
```

23. On the HF view the **transforms.conf** for **splunk_ingest_actions** deployment app.

```
more splunk_ingest_actions/local/transforms.conf
```

You should see the same entries as was listed on the deployment server:

```
[_rule:Badge_Access_US:filter:regex:xxxxxxxx]
INGEST_EVAL = queue=if(match(_raw, "London"), "nullQueue", queue)
STOP_PROCESSING_IF = queue == "nullQueue"
```

24. Exit the HF using the **exit** command.

Task 3: Create data input on UF2 for history_access.log file

25. In Splunk Web on your deployment server, launch the **Settings > Add Data** wizard and add a **Forward** input to monitor **`history_access.log`** on UF2 (**`10.0.0.100`**). Send the data to the **`securityops`** index.

    - On the **Select Forwarders** step:

Selected Server Class	**New**
Selected host(s)	**`LINUX|ip-10-0-0-100`**
New Server Class Name	**`eng_badge_access`**

    - On the **Select Source** step, select **Files & Directories**:

File or Directory	**`/opt/log/badgesv1/history_access.log`**

    - On the **Input Settings** step:

Source type	**Select**, using **Custom > `badge_access`**
Index	**`securityops`**

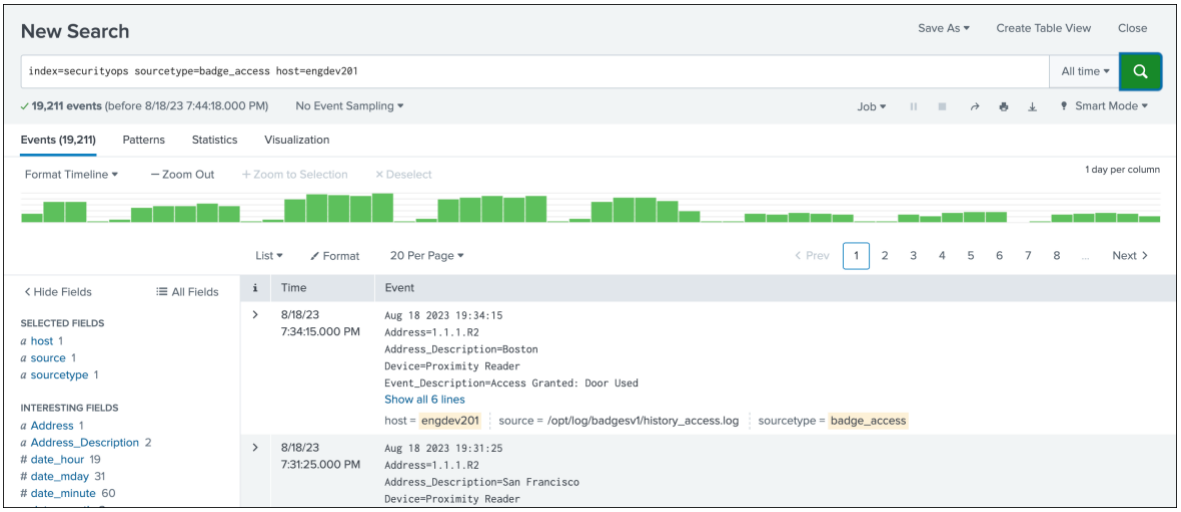
26. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	**`eng_badge_access`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log/badgesv1/history_access.log`**
Includelist	**`N/A`**
Excludelist	**`N/A`**
Source Type	**`badge_access`**
Index	**`securityops`**

27. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID) to verify data has been ingested:

    `index=securityops sourcetype=badge_access host=engdev2##`

    Verify that all the events show an **`Address_Description`** location that is not **`London`**.



NOTE that it can take a few minutes for the data to display on the search head.

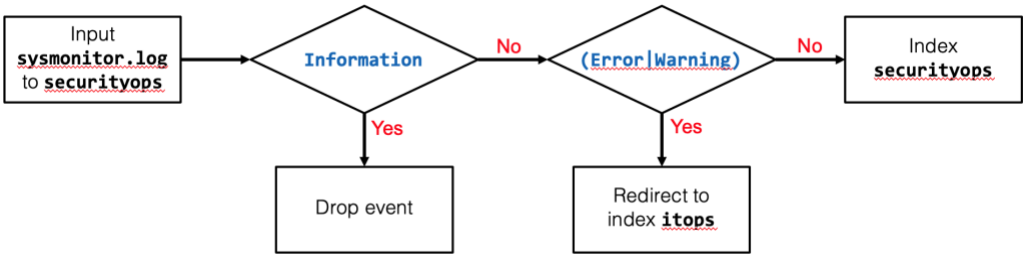
28. Add the text **London** to the search and confirm that no entries for London exist:

```
index=securityops sourcetype=badge_access host=engdev2## London
```

You should receive “No results found”.

Description

Now you will configure the heavy forwarder to perform additional event-level data transformations. All of the sample data contains one of these five values: **Error**, **FailureAudit**, **Information**, **SuccessAudit**, or **Warning**. The task is to configure **props.conf** and **transforms.conf** to allow Splunk to drop or redirect individual events based on REGEX pattern matches as depicted here:



Steps

Task 4: Create a data routing transformation.

In this task, you create transformations to take the following actions:

- If an event contains the regex pattern **Information**, then route to the **nullQueue**.
- If an event contains the regex pattern **(Error|Warning)**, then set index to **itops**.
- Otherwise, for all other events, set the index **securityops** index.

29. On the deployment/test server view the sysmonitor.log file with a text editor or using the **more** command.

```
more /opt/log/adldapsv1/sysmonitor.log
```

```
more C:\opt\log\adldapsv1\sysmonitor.log
```

Review the event data. Notice that the first line is a comma-separated list that defines the columns of data, and that the rest of the file is in the format of a comma-separated values (**csv**) file.

Additionally, each event contains one of five keywords (under the **Type** column): **Error**, **FailureAudit**, **Information**, **SuccessAudit**, or **Warning**:

```
Time,EventCode,EventType,Type,ComputerName,LogName,RecordNumber
"2020-09-10T09:54:14.000-0400",40961,2,Warning,HOST0201,System,770435184
"2020-09-10T09:54:17.000-0400",552,8,SuccessAudit,"BUSDEV-001",Security,880164029
"2020-09-10T09:55:15.000-0400",26,4,Information,HOST0167,System,412563225
"2020-09-10T09:56:14.000-0400",537,16,FailureAudit,"BUSDEV-001",Security,956743389
"2020-09-10T10:53:12.000-0400",17,1,Error,HOST0167,System,836459770
...
```

Quit the **more** command with ‘**q**’ or close the file when done.

30. Edit the **transforms.conf** file using a text editor.

```
/opt/splunk/etc/apps/search/local/transforms.conf
```

```
"C:\Program Files\Splunk\etc\apps\search\local\transforms.conf"
```

31. Append to the current **transforms.conf** file by adding the following stanzas to filter and route events:

```
[eventsDrop]
REGEX = Information
DEST_KEY = queue
FORMAT = nullQueue

[eventsRoute]
REGEX = (Error|Warning)
DEST_KEY = _MetaData:Index
FORMAT = itops
```

The **eventsDrop** stanza looks for the regex pattern **Information**, then routes to the **nullQueue**.

The **eventsRoute** stanza looks for the regex pattern **(Error|Warning)**, then sets the index to **itops**.

For all other events, the index used will be the index that you will later set for the monitor input, which will be the **securityops** index.

32. Edit the **props.conf** file using a text editor:

```
/opt/splunk/etc/apps/search/local/props.conf
```

```
"C:\Program Files\Splunk\etc\apps\search\local\props.conf"
```

33. Append the following stanza to invoke the filtering and routing transformations for the **win_audits** sourcetype:

```
[win_audits]
INDEXED_EXTRACTIONS = csv
TRANSFORMS-information = eventsDrop
TRANSFORMS-itops = eventsRoute
```

The **TRANSFORMS** entries define the actions desired, as defined by the **transforms.conf** stanzas.

The **INDEXED_EXTRACTIONS** entry is added to allow Splunk to properly index this log as a comma-separated values file.

34. Restart the deployment server.

```
/opt/splunk/bin/splunk restart
```

```
C:\Program Files\Splunk\bin\splunk restart
```

Task 5: Add a local file monitor input to test the transformations.

35. Log back into Splunk Web on the deployment server and launch the **Settings > Add Data** wizard.
36. Add a **Monitor** (local) input and select **Files & Directories**.
37. Click **Browse** and navigate to the following file then click **Select**:

```
/opt/log/adldapsv1/sysmonitor.log
```

```
C:\opt\log\adldapsv1\sysmonitor.log
```

38. Verify **Continuously Monitor** is selected and click **Next**.
39. Click Source type: default and type: "**win**"

    The **win_audits** source type should display.

40. Select **win_audits**.

NOTE: If **win_audits** is not available, then make sure you have completed Task 4, Steps 31 and 32 correctly. If you did, go to the address bar on your browser and enter the URI for your **deployment-server/debug/refresh**, click **refresh**, and repeat Steps 7-12. You can also try to restart Splunk to display **win_audits**.

You should see events listed, one per entry in the log file:

Table ▾ Format 20 Per Page ▾ < Prev 1 2 3 4 5 6 7 8 ... Next >

	_time	ComputerName	EventCode	extracted_EventType	LogName	RecordNumber	Time	Type
1	8/4/23 3:54:23.000 AM	HOST0201	40961	2	System	122435838	2023-08-04T06:54:23.000-0400	Warning
2	8/4/23 3:54:24.000 AM	BUSDEV-007	628	8	Security	468983050	2023-08-04T06:54:24.000-0400	SuccessAudit
3	8/4/23 3:54:27.000 AM	BUSDEV-006	624	8	Security	216568644	2023-08-04T06:54:27.000-0400	SuccessAudit
4	8/4/23 3:55:23.000 AM	HOST0167	17	1	System	498008787	2023-08-04T06:55:23.000-0400	Error

41. Click **Next**. From the **Input Settings** step, select **securityops** in the **Index** field.
42. Click **Review** and verify that your input matches the following before clicking **Submit**.

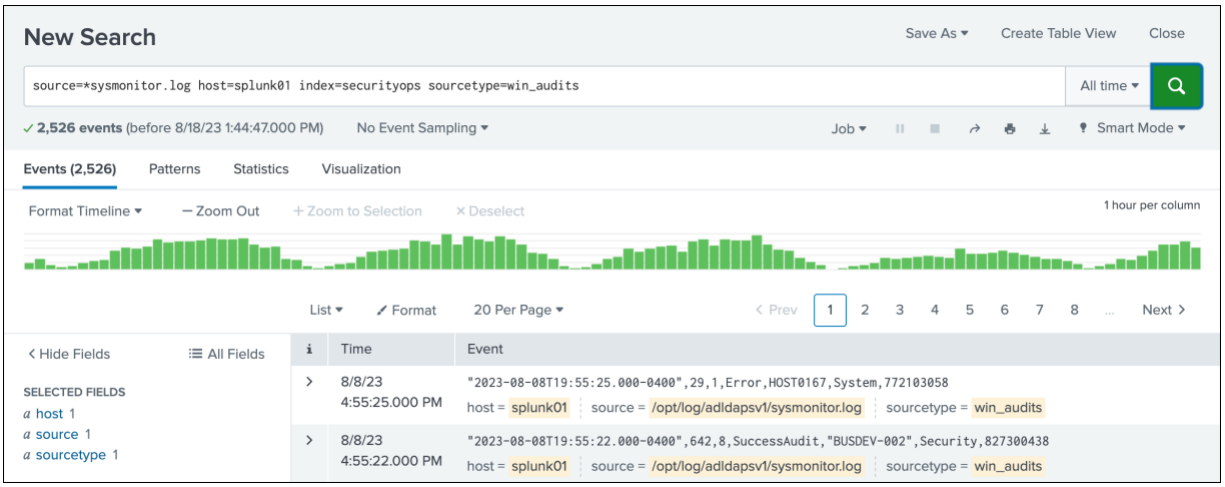
Input Type	**File Monitor**
Source Path	**/opt/log/adldapsv1/sysmonitor.log** **C:\opt\log\adldapsv1\sysmonitor.log**
Continously Monitor	**Yes**
Source Type	**win_audits**
App Context	**search**
Host	**splunk##**
Index	**securityops**

Check Your Work

Task 6: Make sure events are being filtered and routed properly.

43. From the deployment/test server, execute the following search over **All Time**, replacing the ## with your student ID:

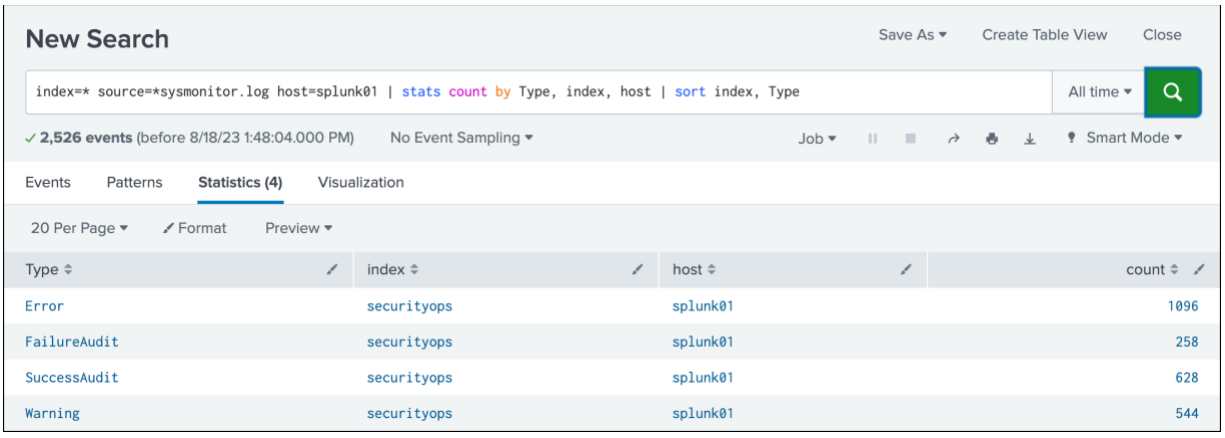
```
source=*sysmonitor.log host=splunk## index=securityops sourcetype=win_audits
```



This search shows events over time sent to the **securityops** index.

44. From the deployment/test server, execute the following search over **All Time**, replacing the ## with your student ID:

```
index=* source=*sysmonitor.log host=splunk##
| stats count by Type, index, host | sort index, Type
```



This search shows the event **Type** and which index it was sent to, as well as the event count. You should not see any "Information" events.

Task 7: Copy the new props and transforms file definitions to the hf_base app and deploy to the HF.

45. On the deployment/test server, copy the contents of the **props.conf** file to the **hf_base** directory.

```
cp /opt/splunk/etc/apps/search/local/props.conf /opt/splunk/etc/deployment-apps/hf_base/local/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
xcopy /S /I /E "C:\Program Files\Splunk\etc\apps\search\local\props.conf" "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

NOTE: The **cp** or **xcopy** command should each be typed all on one line.

46. Copy the contents of the **transforms.conf** file to the **hf_base** directory.

```
cp /opt/splunk/etc/apps/search/local/transforms.conf /opt/splunk/etc/deployment-apps/hf_base/local/
```

Use the Windows file browser to copy the entire directory content. Or, run:

```
xcopy /S /I /E "C:\Program Files\Splunk\etc\apps\search\local\transforms.conf" "C:\Program Files\Splunk\etc\deployment-apps\hf_base\local"
```

NOTE: The **cp** or **xcopy** command should each be typed all on one line.

47. Reload the deployment server. (Splunk may ask you to login as the **admin** Splunk user).

```
/opt/splunk/bin/splunk reload deploy-server
```

```
C:\Program Files\Splunk\bin\splunk reload deploy-server
```

48. From your deployment/test server, connect to HF (**10.0.0.77**):

After establishing an SSH session to your deployment/test server, use SSH to connect to HF (**10.0.0.77**). Log in using your assigned password.

```
ssh {os-user}@10.0.0.77
```

Open the **PuTTy** application, click on session "**SSH to HF**" and click **Open** to start the session. Log in using your assigned password.

49. Confirm the new **[win_audits]** stanza appears with the other stanzas in the deployed **props.conf** file.

```
more ~/splunk/etc/apps/hf_base/local/props.conf
...

[win_audits]
INDEXED_EXTRACTIONS = csv
TRANSFORMS-information = eventsDrop
TRANSFORMS-itops = eventsRoute
...
```

50. Confirm the new **[eventsDrop]** and **[eventsRoute]** stanzas appear in the deployed **transforms.conf** file.

```
more ~/splunk/etc/apps/hf_base/local/transforms.conf

...

[eventsDrop]
REGEX = Information
DEST_KEY = queue
FORMAT = nullQueue

[eventsRoute]
REGEX = (Error|Warning)
DEST_KEY = _MetaData:Index
FORMAT = itops
```

Task 8: Deploy a file monitor to UF2 to transmit the sysmonitor.log data.

51. In Splunk Web on your deployment/test server, launch the **Settings > Add Data** wizard and add a **Forward** input to monitor **sysmonitor.log** on UF2. Send the data to the **securityops** index.

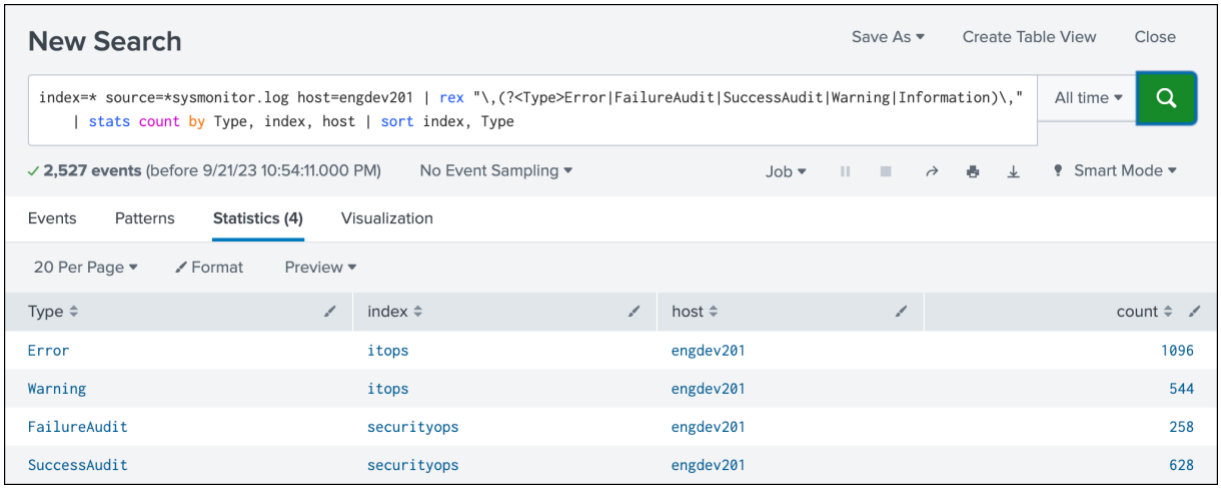
- On the **Select Forwarders** step:
  Selected Server Class **New**
  Selected host(s) **LINUX|ip-10-0-0-100**
  New Server Class Name **eng_sysmonitor**
- On the **Select Source** step, select **Files & Directories**:
  File or Directory **/opt/log/adldapsv1/sysmonitor.log**
- On the **Input Settings** step:
  Source type **Select**, under **Select source type** type "**win**" and select **win_audits**
  Index **securityops**

52. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	**eng_sysmonitor**
List of Forwarders	**LINUX|ip-10-0-0-100**
Input Type	**File Monitor**
Source Path	**/opt/log/adldapsv1/sysmonitor.log**
Includelist	**N/A**
Excludelist	**N/A**
Source Type	**win_audits**
Index	**securityops**

53. In Splunk Web on the search head execute the following search over the **All time** (replace the ## with your student ID):

```
index=* source=*sysmonitor.log host=engdev2##
| rex "\,(?<Type>Error|FailureAudit|SuccessAudit|Warning|Information)\,"
| stats count by Type, index, host | sort index, Type
```



It may take a few minutes for the results to appear.

NOTE: An additional Splunk search line with the **rex** command was necessary on the Search Head, which was not needed when running this search on the Deployment/Test Server:

```
index=* source=*sysmonitor.log host=engdev2##
| rex "\,(?<Type>Error|FailureAudit|SuccessAudit|Warning|Information)\,"
| stats count by Type, index, host | sort index, Type
```

The indexed extraction with a comma-separated values (CSV) was configured on the Deployment/Test Server, but not in the Splunk production environment indexers.

Troubleshooting Suggestions

If your searches are not producing the expected results, check your configurations.

1. Verify the syntax and spelling in all configurations and searches.
2. If you make any corrections, clear the fishbucket checkpoint for **/opt/log/adldapsv1/sysmonitor.log** on the forwarder:

```
./splunk stop

./splunk cmd btprobe -d ~/splunkforwarder/var/lib/splunk/fishbucket/splunk_private_db
   --file /opt/log/adldapsv1/sysmonitor.log --reset

./splunk start
```

NOTE: The **btprobe** command should be typed all on one line.

Module 15 Lab – Supporting Knowledge Objects

Description

In this exercise, you will create a knowledge object (a report) as the **admin** user and reassign it to another user, **emaxwell**.

Steps

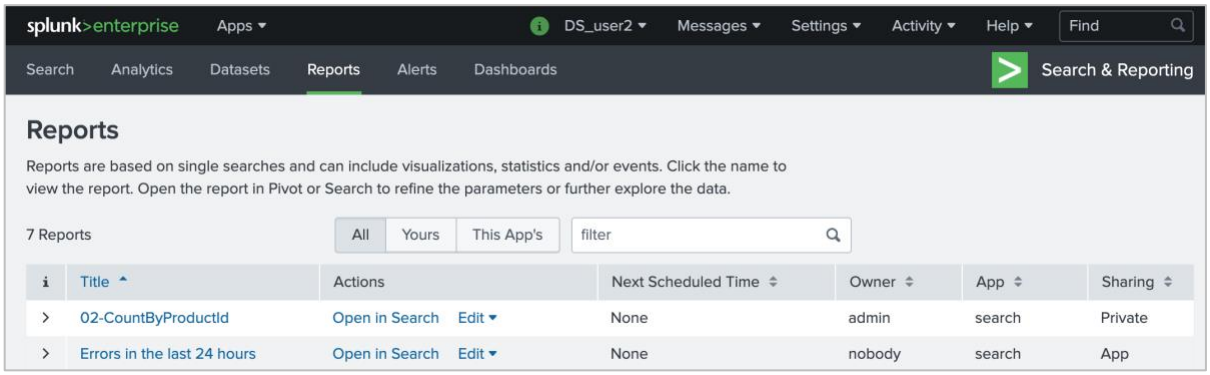
Task 1: Log into the deployment/test server and create a knowledge object (report).

1. Log into the deployment/test server as **admin**.
2. From the deployment/test server, execute the following search for the **Last 24 hours**:

   ```
   index=test sourcetype=access* | stats count by productId
   ```

3. Click **Save As** > **Report**.
4. In the **Title** field, name your report **##-CountByProductId** replacing the ## with your student ID.
5. Click **Save** and then, click **View**.

6. In the Splunk Web black menu bar, click **Apps > Searching and Reporting**.
7. In the grey menu bar, click **Reports**.

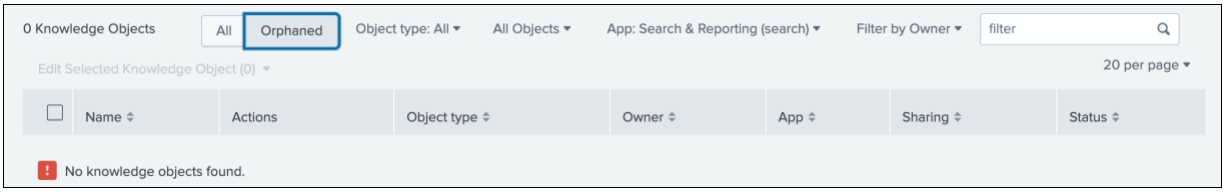


8. Click on and view the report you just created.

Task 2: Look for orphaned knowledge objects.

9. Click **Settings > All configurations**.
10. Click the **Reassign Knowledge Objects** button in the top right.
11. From the **All | Orphaned** button, click **Orphaned**.
12. Select **Filter by Owner > All**.

    You should see **No knowledge objects found** indicating no orphaned knowledge objects.



Task 3: Reassign the orphaned report to emaxwell.

13. From the **All | Orphaned** button, click **All**.
14. In the **filter** text box to the right of the **Filter by Owner** drop-down list, type your 2-digit student ID number, and press **Enter**.

    You should see your report listed.

15. Click the **Reassign** link under the **Actions** column.
16. From the **Reassign Entity** dialog box, click the **New Owner** dropdown.
17. Select **`(emaxwell)`** and click **Save**.

Task 4: Verify you no longer can see the report.

18. Click on **Apps** dropdown and select **Searching and Reporting**, then click **Reports**.

    You should ***not*** see your report listed.

19. Click **Yours**.

    You should ***not*** see your report listed.

Task 5: Log into the deployment/test server and verify knowledge object assignment.

20. Log out from Spunk Web, and log back into the deployment/test server as **`emaxwell`** (password **`open.sesam3`**)
21. On the left pane titled **Apps**, click on **Search & Reporting** (click **Skip/Skip tour** if any messages appear), then click **Reports**.

    You should see your report listed.

22. Click **Yours**.

    You should see your report listed.

23. Click your report to run and test it.