splunk>

Splunk Enterprise 8.0 Data Administration – Class Lab Exercises

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
{user-ID}	Your assigned Splunk username
{password}	Your assigned password
{DS-eip}	The external IP address of your assigned deployment server
{DS-iip}	The internal IP address of your assigned deployment server
{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:

ip-10-0-0-50	The host name of your Splunk universal forwarder #1 (UF1). It has the private address of **10.0.0.50**.
ip-10-0-0-100	The host name of your Splunk universal forwarder #2 (UF2). It has the private address of **10.0.0.100**.
ip-10-0-0-77	The host name of your Splunk Heavy Forwarder. It has the private address of **10.0.0.77**.

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 Forwarders:	**/opt/home/{os-user}/splunkforwarder**

The following text editors are installed in your environment:

Linux server:	**nano** **vi**
Windows server:	**Notepad++**

If you are unfamiliar with **vi**, use **nano**. It is an easy text editor.

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


Linux OS


Windows OS

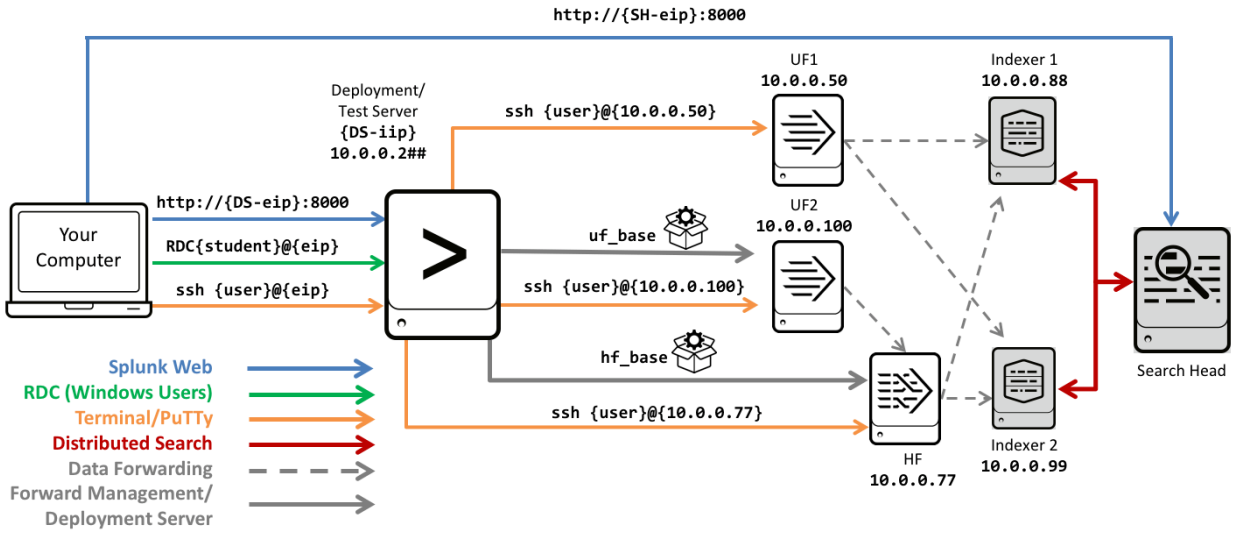
NOTE:	When you access the Splunk Search app user interface the first time, Splunk asks if you want a tour of the app. Throughout the exercises, you may dismiss this prompt at any time.

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>

Module 1 Lab Exercise – Explore your Splunk Lab Environment

Welcome to the Splunk Education lab environment. In this exercise, you will perform basic configuration tasks using the Splunk Web interface and, using the CLI, investigate Splunk system 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 uniquely identify your work from other class participants' work. Substitute the "##" references in this lab document with your student ID, when asked to.

Student ID: _____
{student-ID}

Search Head Credentials

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

Search Head Splunk Web URL: **http://_____:8000**
{SH-eip}

Splunk username: _____ Password: _____
{user-ID} **{password}**

Deployment Server/Test Server Credentials

You have been assigned your own deployment server/test server Splunk instance. The command line access procedure depends upon the underlying operating system (Linux or Windows). Splunk Web (browser) access procedures are the same regardless of the underlying operating system.

Deployment Server/Test Server Splunk Web URL: **http://_____:8000**
{DS-eip}

Splunk username: **admin** Password:_____
{password}

Linux OS

To access the Linux operating system, you will use an SSH client such as Mac Terminal or PuTTY (Windows).

Linux host IP address name: _____
{DS-eip}

Linux Username: _____ Password: _____
{os-user} **{password}**

Windows OS

To access the Windows operating system, you will use a Remote Desktop client (RDC), such as Microsoft Remote Desktop.

Windows host IP address name: _____
{DS-eip}

RDC Username: **student** Password: **splunk3du**

splunk>

Lab Environment Discovery Steps

Task 1: Access Splunk Web on the Search Head.

You will access the shared search head and your personal deployment/test server instances frequently with Splunk Web throughout the lab exercises. It is strongly recommended that you keep a separate tab or window open to each machine so you can context-switch easily between them when necessary. 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. Another option is to use two different web browsers. For example, use Chrome to access your search head and Firefox to access the deployment test server. A third option is to change the color of the search app navigation bar. Your instructor may have already done this for the shared search head.

1. Navigate to the search head (using your browser of choice).
2. Log in with your assigned **`{user-ID}`** and password **`{password}`**.
3. From the Splunk bar, to identify the Splunk version that the search head is running, click **Help > About**.
4. From the Splunk bar, click your **`{SH_user-ID}`** name.
5. Click **Account Settings**.
6. In the **Full name** field, notice your name preceded by **SH_**. (This identified your login session and the search head.) Do not change.
7. 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**.

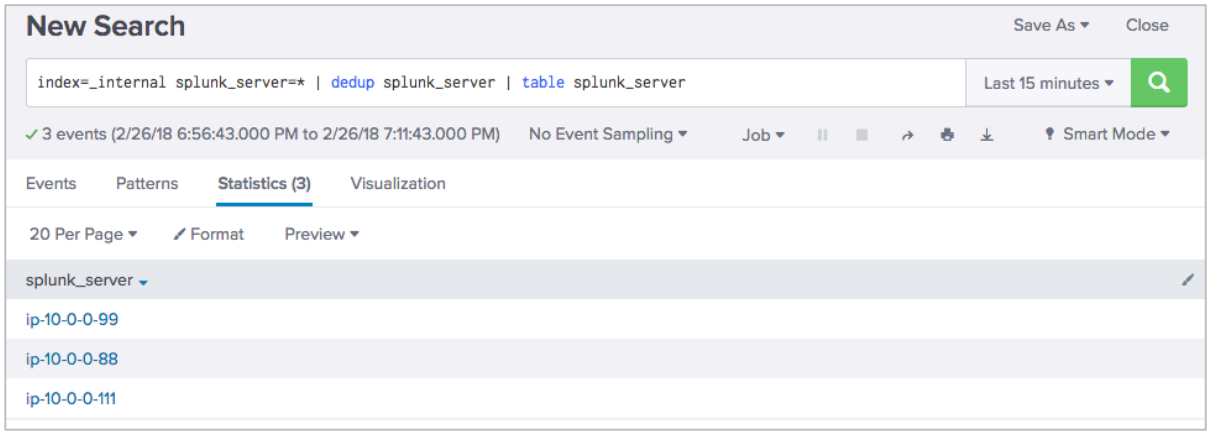
NOTE: Do not change your assigned password.

8. From the Splunk bar, click your **`{SH_user-ID}`** name and click **Preferences**.
9. In the **Default** application field, select **Search & Reporting**.
10. Click **Apply**.
11. In the app navigation bar, click **Apps > Search & Reporting**.
12. Click **Skip** to dismiss the tour message.
13. Click **Settings**. The options shown are the defaults available to the Splunk **power** role.

Task 2: Run a search on the Search Head.

14. 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 out-of-the-box 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.

ip-10.0.0.111 – shared search head
ip-10.0.0.88 – shared Indexer 1
ip-10.0.0.99 – shared Indexer 2

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 server/test server to change server settings.

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

    http://{DS-eip}:8000

16. Log in as **admin** using your assigned password {**password**}.
17. Click **Skip** to dismiss the **Help us improve Splunk software** message, if it appears.
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. Notice your assigned user ID in the Splunk bar with a DS_ prefix. This identifies your login session and the deployment/test server. **Do not change**.
20. Click **{DS_user-ID} > Preferences**.
21. The **Global** setting should be selected, change the **Time zone** to your current time zone.

22. Click **Apply**.
23. Click **{DS_user-ID} > Account Settings**.

    Notice in the **Full name** field your assigned **{DS_user-ID}**. **Do not change**.
24. In the **Email address** field, notice your two-digit **{student-ID}** (leading zero required for student IDs 01-09). **Do not change**.
25. Navigate to **Settings > Server settings > General settings**.

    The directory where Splunk is installed is referred to as **SPLUNK_HOME**. Record the path specified in the **Installation path** field here: _____ .
26. Rename the **Splunk server name** and **Default host name** using the following convention:

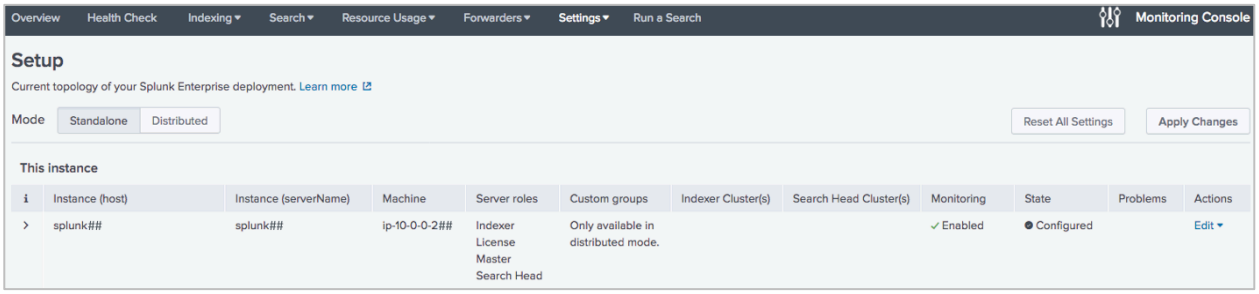
    Splunk server name: **splunk##** where ## is your **{student-ID}**

    Default host name: **splunk##** where ## is your **{student-ID}**
27. Click **Save**.

    Notice the message. Splunk must be restarted for changes to take effect.
28. Click **Messages > Click here to restart from Server controls > Restart Splunk > OK**.
29. After the restart, click **OK** and log back into Splunk Web.

Task 4: View the changes made to the deployment/test server.

30. To access the monitoring console, click **Settings > Monitoring Console**.
31. From the app navigation menu, click **Settings > General Setup**.
32. Verify the server name and note the discovered default server roles.



33. Click **Edit > Edit Server Roles**. (You may need to scroll to the right in the table to see the **Edit** hyperlink, depending on the size of your browser window.)
34. Remove the check mark from **Search Head**, and select the check mark for **Deployment Server**, then click **Save > Done**.
35. To complete the app setup, click **Apply Changes > Go to Overview**.
36. 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: Retrieve Splunk settings from your deployment server using the CLI.

37. Connect to your deployment server's command line as follows:

Linux Splunk server, use Terminal (Mac) or PuTTY (Windows)

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

Windows Splunk server, use **Remote Desktop Connection**

- Start Remote Desktop Connection and enter **{DS-eip}.**
- User name is **student** and your assigned password.

38. Navigate to the **SPLUNK_HOME/bin** (documented previously in Task 3, step 25.). For example:

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

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

39. Use **splunk help commands** and **splunk help show** to obtain a list of Splunk CLI commands and syntax help.

40. Execute the following CLI commands to check the status of your Splunk server and determine other details.

```
./splunk status
```

```
splunk status
```

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

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

The commands shown below assume that you are already in the **SPLUNK_HOME/bin** directory. Some commands will require you to log in as **admin** before executing. (user: **admin**, password: **{password}**, which is likely: **splunk3du**)

```
Your session is invalid.  Please login.
Splunk username: admin
Password:
```

If you are on a Windows server, remove **./** from the commands. For example, type **splunk version**, instead of **./splunk version**

Splunk version **./splunk version**

Splunk Web port: **./splunk show web-port** returns **8000**

Splunk management (splunkd) port: **./splunk show splunkd-port** returns **8089**

Splunk App Server ports: **./splunk show appserver-ports** returns **8065**

Splunk KV store port: **./splunk show kvstore-port** returns **8191**

Splunk server name: **./splunk show servername** returns **splunkXX**

Default host name: **./splunk show default-hostname** returns **splunkXX**

Troubleshooting Suggestions

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
```

splunk>

Module 2 Lab Exercise – Add a Local Data Input

Description

In this lab exercise, you will create all the local indexes on the deployment/test server required for the subsequent lab exercises. In later lab exercises, you will forward data inputs to remote indexers. This requires that your deployment/test server have the same local indexes as the remote indexers. Finally, you will create a local file input (monitor) to be indexed on your deployment/test server.

Steps

Task 1: Create the local indexes on the deployment/test server.

1. Access Splunk Web on the deployment/test server, click **Settings > Indexes**.
2. Click **New Index**.
3. Populate the form as follows:

Index Name:	`test`
Index Data Type:	`Events` (default)
App:	`Search & Reporting` (This saves the configurations within the Search app-context).

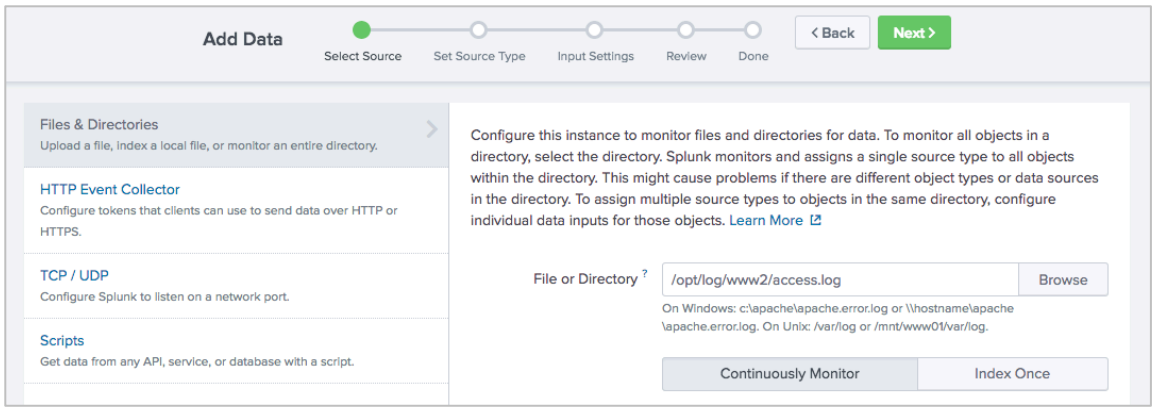
   Leave the rest of the fields empty to accept the defaults.
4. Click **Save**.
5. Repeat steps 1 through 5 to create the following indexes:
   - `itops`
   - `sales`
   - `securityops`
   - `websales`

Task 2: Index events from an access.log file to a test index.

6. Click **Settings > Add Data**.
7. If you see the **Welcome**, **Administrator** tour message, click **Skip**.
8. Click **Monitor** to launch the **Add Data** wizard.
9. On the **Select Source** step, click **Files & Directories**.
10. 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`



11. 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.)

12. 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.

13. 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)

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

15. 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`**

16. Click **Submit**.

Check Your Work

Task 3: Confirm your input configuration.

17. To verify your monitor input, click **Start Searching**.
18. Click **Skip** to dismiss any message that may appear.
19. Observe the search string:

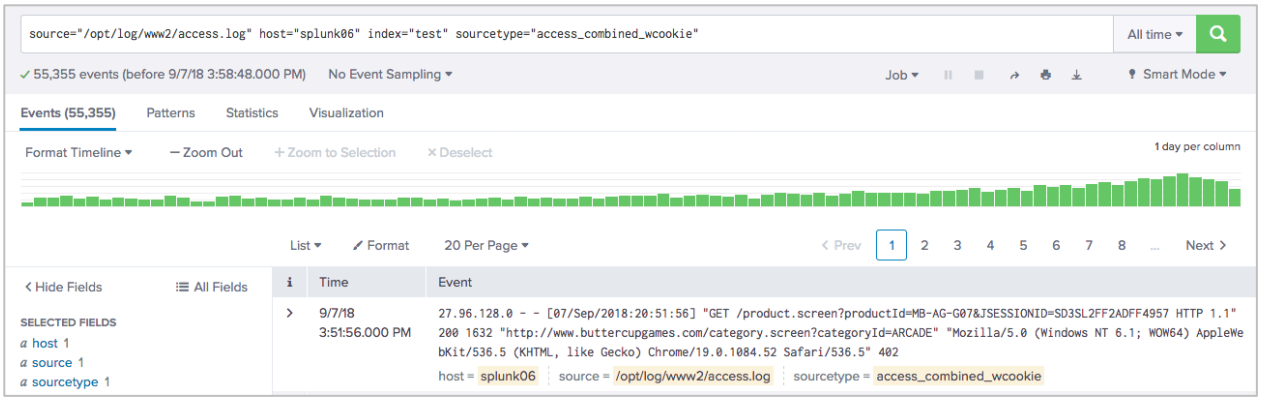
    Linux server:

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

    Windows server:

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

20. Observe the automatically extracted field names and values:



Task 4: View the input stanza.

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

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

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

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.

22. Use the **btool** command to show all of the Splunk settings associated with the creation of the data input.

NOTE: Remember in these labs to navigate to the **SPLUNK_HOME/bin** directory to run the **splunk** command. For example:

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

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

```
./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
```

NOTE: The **_rcvbuf** field shows the receive buffer default used for UDP port input. The **host** field shows the default hostname as defined in **$SPLUNK_HOME/etc/system/local/inputs.conf** (on Linux) or **C:\SPLUNK_HOME\etc\system\local\inputs.conf** (on Windows).

23. 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 = splunk02
/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
```

splunk>

Module 3 Lab Exercise – Setting Up Forwarders

Description

In this exercise, you configure universal forwarder #1 (UF1, **10.0.0.50**) to send data to the remote indexers (**10.0.0.88** and **10.0.0.99**) and validate the receipt of internal splunkd data on the shared search head.

Task 1: Connect to Universal Forwarder #1 (UF1).

1. Connect to the UF1 (**10.0.0.50**) using the following OS-specific instructions:

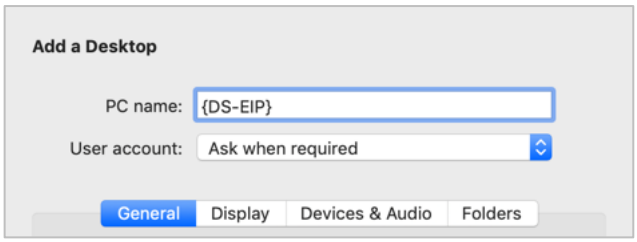
Use SSH to connect to your Linux depoyment/test server using IP address represented by **{DS-EIP}**.

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

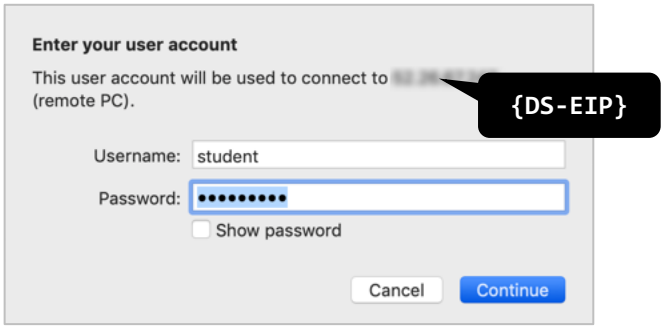
After establishing an SSH session to your deployment/test server, use SSH to connect to UF#1 (**10.0.0.50**).

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

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}**.



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



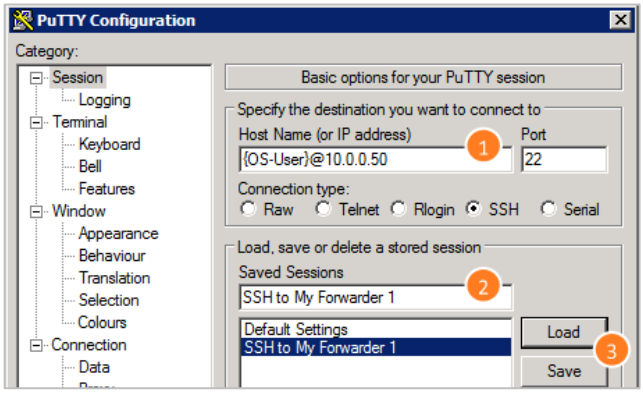
After connecting to your deployment/test server, locate **PuTTy** on the desktop:

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

a. ❶ Replace **`{os-user}@10.0.0.50`** with your designated values.

b. ❷ Name your session and ❸ Save.

c. Click **Open** to start the session.



2. Click **Yes** to accept the server's host key and enter your password. After connected to UF1 (**`10.0.0.50`**), the command prompt indicates the location:

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

Task 2: Start and configure your forwarder instance.

3. 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>`**.

4. Using the **`show`** command, view the **splunkd-port** number (Splunk will prompt you for the **`admin`** username and password which is **admin** and **your assigned password**.)

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

5. Using the **set** command, change your forwarder's **servername** and the **default-hostname** to **engdev1{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.
```

6. Restart UF1 to apply your changes.

```
./splunk restart
```

Task 3: Configure your forwarder to send data directly to the 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.

7. 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 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.
```

8. 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 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
```


9. 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
```

10. Restart UF1 to apply your new changes.

```
./splunk restart
```

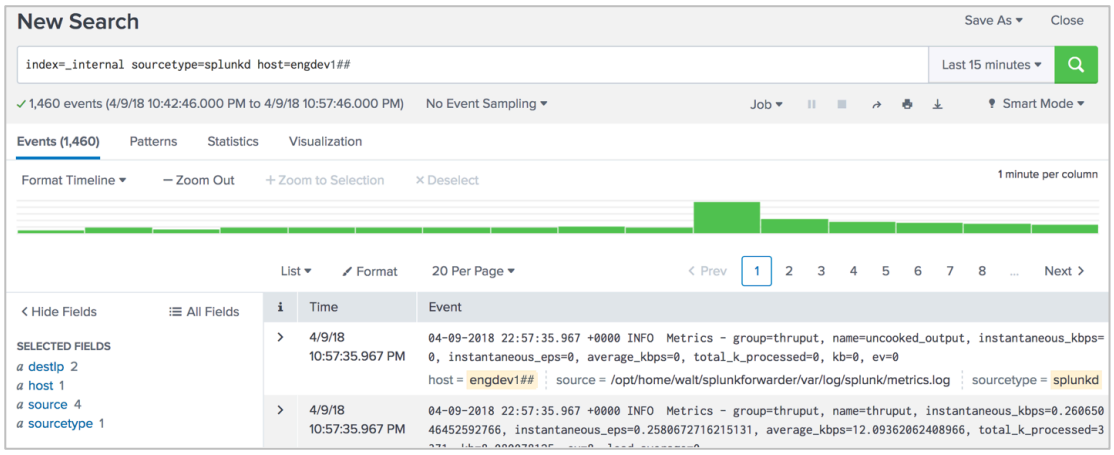
11. Exit UF1's SSH session.

```
exit
```

Task 4: Validate the receipt of forwarded data.

12. Using the search head, 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##
```



13. You should see events related to the splunkd process coming from your UF1.

Module 4 Lab Exercise – Configure Forwarder Management

Description

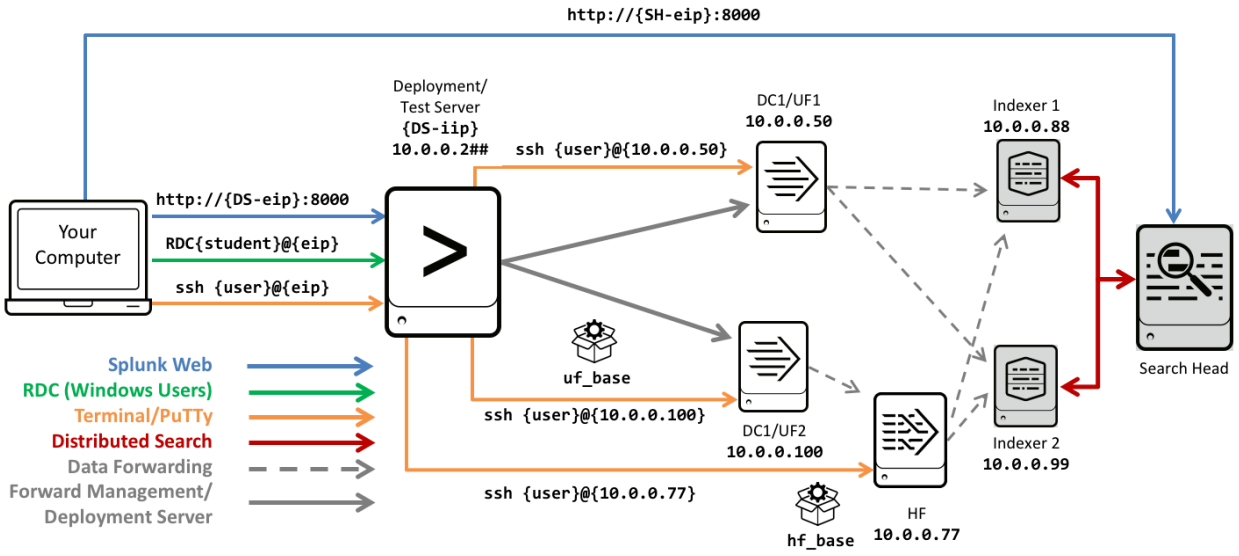
In this exercise, you will use the Forwarder Management interface in Splunk Web to configure a remote universal forwarder and a heavy forwarder. The advantage of this option is that it allows you to manage multiple groups of forwarders from a central location.

First, you will enable the deployment server feature on your deployment server instance and stage two deployable apps for your forwarders that have already been created for you. You will use these apps to configure the **`outputs.conf`** file which is needed 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 launch a second universal forwarder (**`10.0.0.100`**) and a heavy forwarder (**`10.0.0.77`**) and configure them 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: 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.

1. Access your deployment server's command line (SSH for Linux, RDC for Windows).
2. 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"
```

3. Navigate to the **local** directory of the **uf_base** app and list its contents to make sure the **outputs.conf** file was copied successfully.

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.

4. Using a text editor, add the stanza below to the **outputs.conf** file by replacing **##** with your {**student-ID**}. (Linux users can use vi or nano, Windows users can use **Notepad++.)**

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

NOTE: 	**Windows Users**: You must have administrator rights when editing the Splunk configuration files in the lab environment. Use the **Notepad++** icon on the desktop to launch the application with administrator rights. If you are not sure, you can use the following guidelines for opening and saving the files. • 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).

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

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

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

5. Save and close the edited file.

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

In this task, you manually configure the forwarder as a deployment client by using the **`splunk set deploy-poll`** command. Since many Splunk environments use hundreds or thousands of forwarders, this is not practical or scalable. Most Splunk customers use a third-party software configuration management tool, such as Puppet or Chef. Another option is to include the Universal Forwarder software with a **`deploymentclient.conf`** file into pre-configured software builds.

6. From your deployment server's command line, SSH into your UF2 (**`10.0.0.100`**). (Refer to Task 1 of the previous exercise for OS-specific instructions.)

```
os-user@ip-10-0-0-2xx ~]$ ssh os-user@10.0.0.100
os-user@10.0.0.100's password:      (Use your assigned password)
```

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

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

8. Use the CLI to determine the auto-assigned management port number. (Splunk will prompt you for the **`admin`** username and password which is **admin** and **your assigned password**.)

```
./splunk show splunkd-port
Splunkd port: 1##89          (This is the auto-assigned splunkd port. The ## is your student-ID number)
```

9. Using the **`set servername`** and **`set default-hostname`** commands, change your forwarder's server name and default hostname to **`engdev2{student-id}`**:

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

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

```
./splunk set servername engdev2##              (where ## is your student-ID)
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.
```

10. Use the **`set deploy-poll`** command to establish communication between the forwarder and the deployment server, then restart the forwarder.

```
./splunk set deploy-poll 10.0.0.2##:8089        (where ## is your student-ID)
Configuration updated.
You need to restart the Splunk Server (splunkd) for your changes to take effect.

./splunk restart
Stopping splunkd
...

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

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

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

```
./splunk show deploy-poll
Your session is invalid.  Please login.
Splunk username: admin
Password: {passw}                                    (Use your assigned password)
Deployment Server URI is set to "10.0.0.2##:8089"  (where ## is your student-ID number)
```

12. 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
```

```
exit
```

Task 3: 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.

13. Access your deployment server's command line (SSH for Linux, RDC for Windows).
14. 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"
```

15. Navigate to **`local`** directory of the **hf_base** app and list its contents to make sure the **`outputs.conf`** file was copied successfully.

16. Read the contents of the **outputs.conf** file:

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

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

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

17. From your deployment server's command line, SSH into your HF (**10.0.0.77**). (Refer to Task 1 of the previous exercise for OS-specific instructions.)

```
os-user@ip-10-0-0-2xx ~]$ ssh {os-user}@10.0.0.77
os-user@10.0.0.100's password:       (Use your assigned password)
```

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

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

19. Use the CLI to determine the auto-assigned management port number. (Splunk will prompt you for the **admin** username and password which is **admin** and **your assigned password**.)

```
./splunk show splunkd-port
Splunkd port: 1##89              (This is the auto-assigned splunkd port. The ## is your student-ID.)
```

20. Set up the receiving port on your heavy forwarder to receive data from UF2.

```
./splunk enable listen 99##        (where ## is your student ID)
Listening for Splunk data on TCP port 99##.      (where ## is your student-ID)
```


21. Using the **set servername** and **set default-hostname** commands, change your heavy forwarder's server name and default hostname to **splunkHF{student-id}**:

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

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

```
./splunk set servername splunkHF##              (where ## is your student-ID)
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.
```

22. Use the **set deploy-poll** command to establish communication between the forwarder and the deployment server, then restart the forwarder.

```
./splunk set deploy-poll 10.0.0.2##:8089        (where ## is your student-ID)
Configuration updated.
You need to restart the Splunk Server (splunkd) for your changes to take effect.

./splunk restart
Stopping splunkd
...

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

23. Use the **show deploy-poll** command to verify the deployment-client configuration. **10.0.0.2##** is the internal address of your deployment server instance.

```
./splunk show deploy-poll
Your session is invalid.  Please login.
Splunk username: admin
Password: {passw}                               (Use your assigned password)
Deployment Server URI is set to "10.0.0.2##:8089"          (where ## is your student-ID)
```

24. 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

exit
```

Task 5: Add a server class to manage the HF from your deployment server.

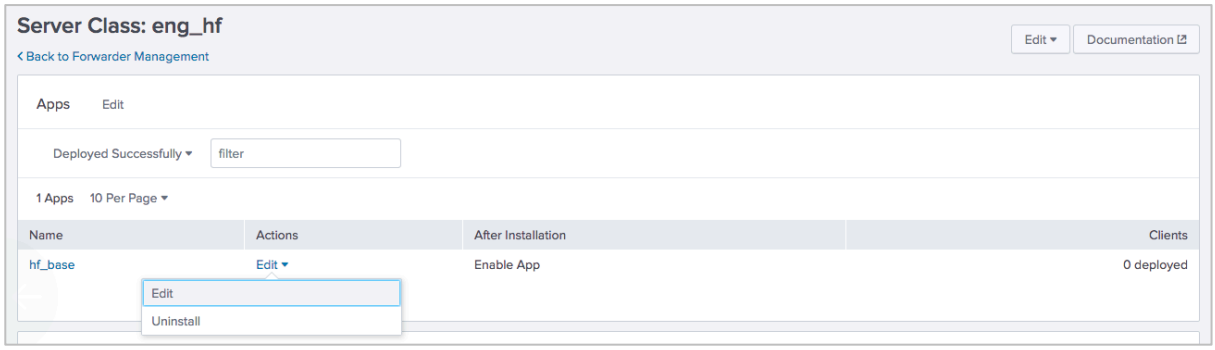
You should now have two deployment apps and two deployment clients running. To complete the forwarder management enablement, you will need to configure a server class for each deployment client and assign the appropriate app. In this task, you will create a server class for the HF client and assign it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

25. Log into Splunk Web as **admin** on the deployment server.
26. Navigate to **Settings > Forwarder management**.
27. Select the **Apps** tab. The **hf_base** and **uf_base** apps should display.
28. Select the **Clients** tab. The host **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.

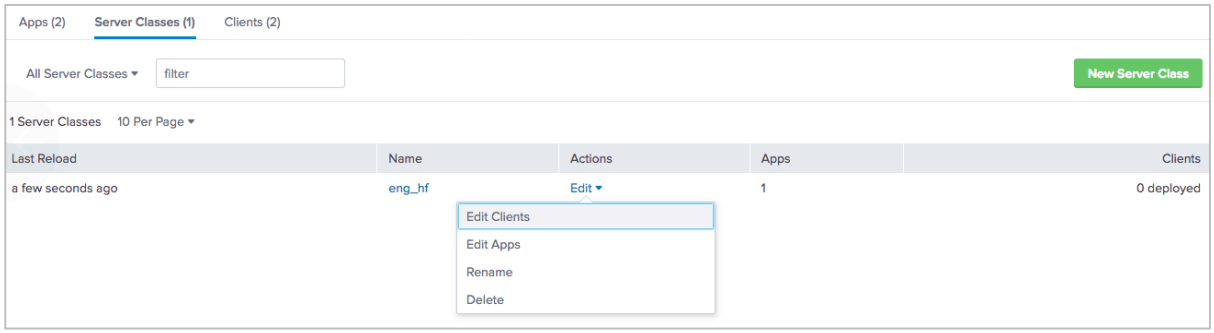
29. On the **Server Classes** tab, click **create one**.
30. In the **New Server Class** window, name the server class **eng_hf** and click **Save**.
31. To add the **hf_base** app to the server class, click **Add Apps**.
32. Click the **hf_base** app to move it to the **Selected Apps** panel, then click **Save**.

    The **After Installation** column of the **hf_base** app currently shows **Enable App**.
33. Click **Edit** > **Edit.**



34. On the **Edit App: hf_base** page, select the **Restart Splunkd** check box, then click **Save**.
35. Select the **Server Classes** tab.

36. In the Actions column, click **Edit > Edit Clients**.

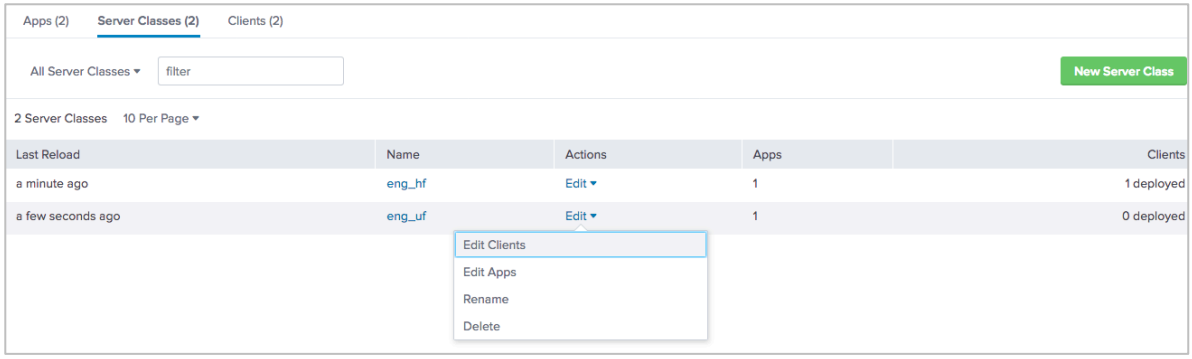


37. Enter the deployment client's IP address **`10.0.0.77`** to the **Include (whitelist)** box.
38. Click **Preview**.
39. When the check mark appears in the **Matched** column, click **Save**.

Task 6: Add a server class to manage UF2 from your deployment server.

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

40. From the **Server Classes** tab, click **New Server Class**.
41. In the **New Server Class** window, name the server class **`eng_uf`** and click **Save**.
42. To add the **`uf_base`** app to the server class, click **Add Apps**.
43. Click the **`uf_base`** app to move it to the **Selected Apps** panel, then click **Save**.
44. In the **After Installation** column of the **uf_base** app, it displays **Enable App**.
45. Click **Edit** > **Edit.**
46. On the **Edit App: uf_base** page, select the **Restart Splunkd** check box, then click **Save**.
47. Click the **Server Classes** tab.
48. In the **Actions** column, click **Edit > Edit Clients** of the **eng_uf** server class.



49. Enter the deployment client's IP address **`10.0.0.100`** to the **Include (whitelist)** box.
50. Click **Preview**.
51. When the check mark appears in the **Matched** column, click **Save**.

Check Your Work

Task 7: Confirm the deployment of the hf_base app.

52. From your heavy forwarder (**10.0.0.77**) terminal window, confirm that the directory **hf_base** exists in the **~/splunk/etc/apps** directory.

```
cd ~/splunk/etc/apps
ls -t
hf_base                          introspection_generator_addon
search                           launcher
...
```

53. Verify that the **outputs.conf** file matches the following, then exit the SSH session.

```
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

exit
```

Task 8: Confirm the deployment of the uf_base app.

54. From your UF#2 (**10.0.0.100**) terminal window, confirm that the directory **uf_base** exists in the **~/splunkforwarder/etc/apps** directory.

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

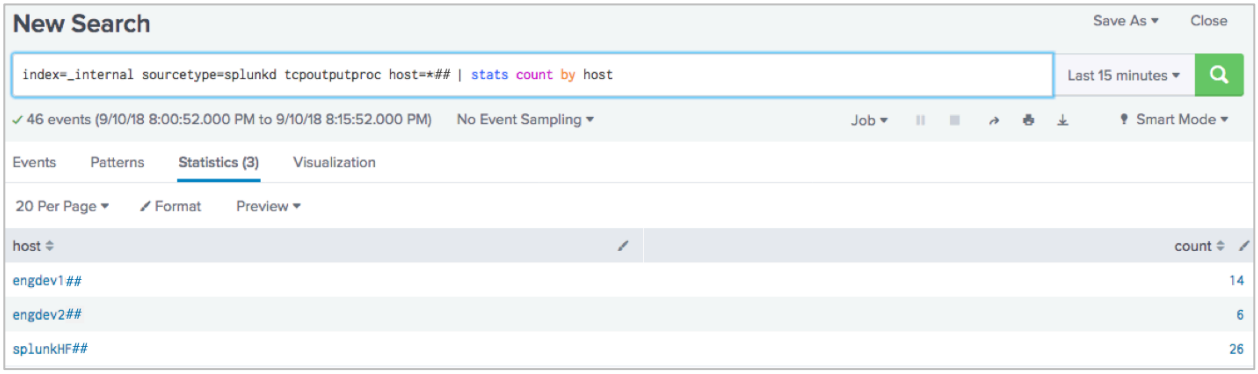
55. Verify that the **outputs.conf** file matches the following, then exit the SSH session.

```
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##

exit
```

56. From the search head, replace the **##** with your student ID and execute the following search over the **Last 60 minutes**:

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

57. You should now see following hosts:
    - **engdev1##** (UF1, ## = student id)
    - **engdev2##** (UF2, ## = student id)
    - **splunkHF##** (Heavy Forwarder, ## = student id)



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 forwards**, 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 5 Lab Exercise – File Monitor Input

Description

In this exercise, 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 indexers (**`IDX1`** and **`IDX2`**). 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

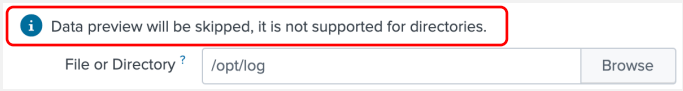
Task 1: Add a test directory monitor input to an index on the Deployment Server.

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.

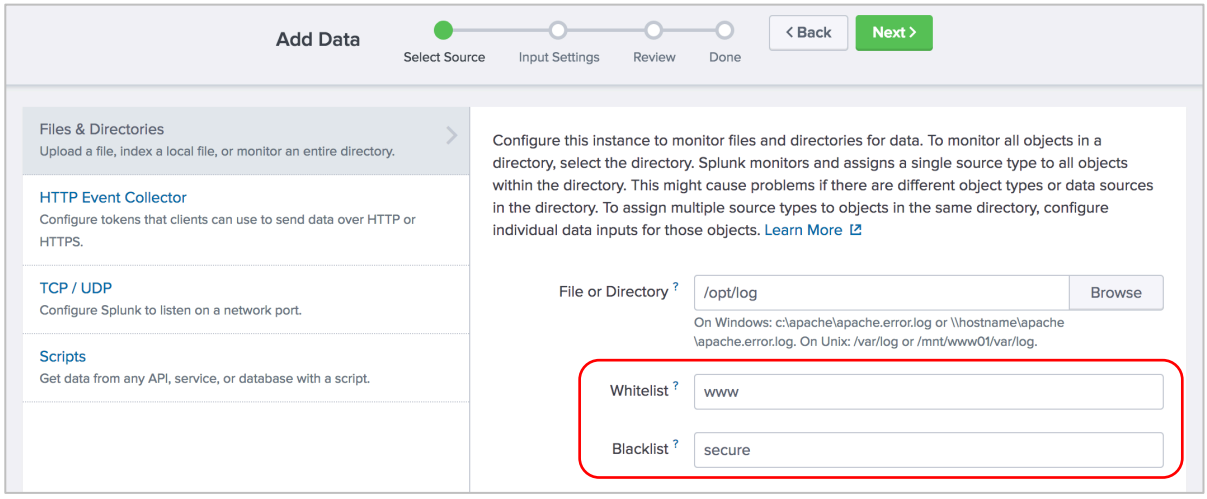
1. On your deployment server, click **Settings > Add Data > Monitor**.
2. On the **Select Source** step, click **Files & Directories**.
3. Click **Browse**, navigate to the directory below and click **Select**.

	`/opt/log`
	`C:\opt\log`

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



4. On the **Select Source** step, type **`www`** for the **Whitelist** and **`secure`** for **Blacklist** and then click **Next**.



5. On the **Input Settings** step, select the follwing options and click **Review**:

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

6. 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)*
Whitelist	**`www`**
Blacklist	**`secure`**
Source Type	**`Automatic`**
App Context	**`search`**
Host	**`splunk##`**
Index	**`test`**

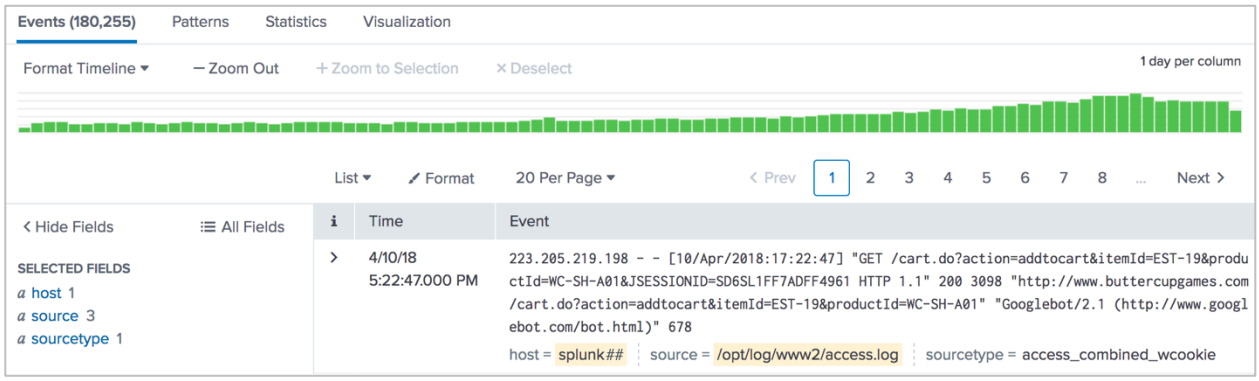
7. Click **Submit**.
8. To verify your monitor input, click **Start Searching**. (If you get a Welcome message, click **Skip**.)
9. Observe the search string (using ## to match your student ID):

   Linux server:

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

   Windows server:

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



10. 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`**

11. From your deployment server, view the **inputs.conf** file and verify the new stanza.

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

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

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

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

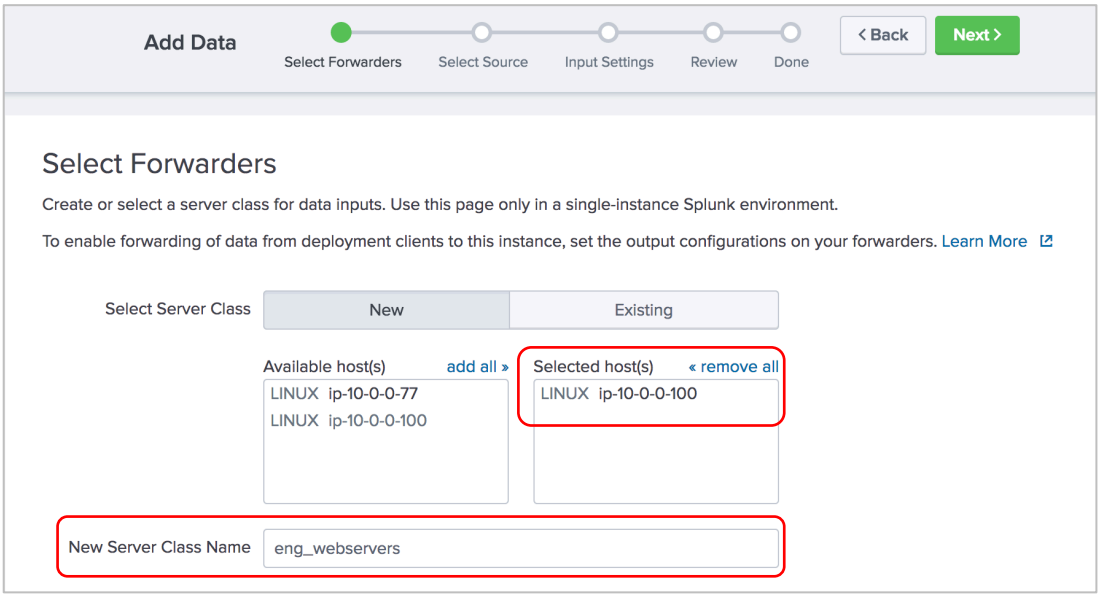
```
[monitor://C:\opt\log\www2\access.log]
disabled = false
index = test
sourcetype = access_combined_wcookie

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

Task 2: Add a directory monitor input to index remote data from UF2.

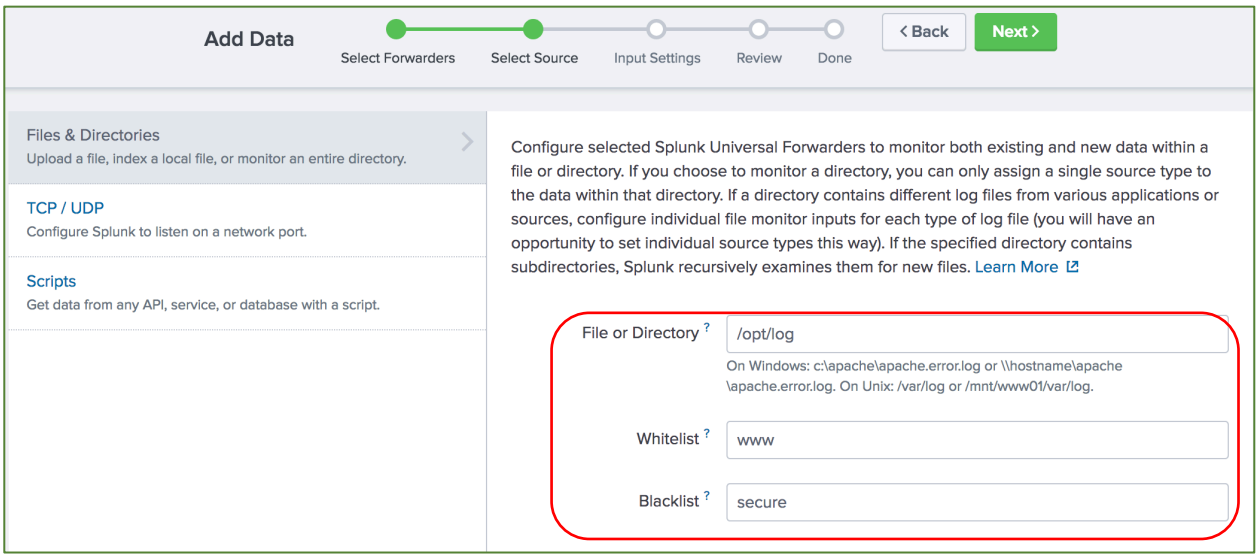
Now that you have successfully indexed a directory monitor input on your test server, you will index the same directories located on UF2 to the remote indexes located on indexers (IDX1 and IDX2). The **Add Data** wizard's **forward** feature automatically creates the **`inputs.conf`** file in a deployable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

12. On your deployment server, click **Settings > Add Data > Forward**.
13. 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`**



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



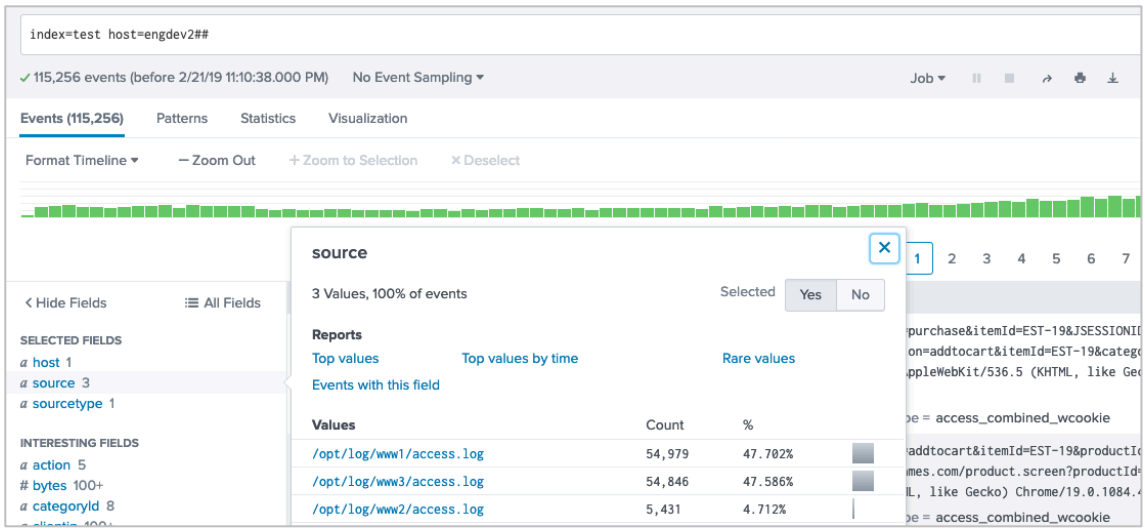
15. For the **Input Settings**, leave the **Source type** as **Automatic** and select **`test`** for the **Index** and click **Review**.
16. Verify your settings match the following, then click **Submit**:

Server Class Name	**`eng_webservers`**
List of Forwarders	**`LINUX|ip-10-0-0-100`**
Input Type	**`File Monitor`**
Source Path	**`/opt/log`**
Whitelist	**`www`**
Blacklist	**`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/heavy forwarder), you would see the local data you indexed in Task 1, not the data from the universal forwarder.

17. Open Splunk Web on your search head. Replace the ## with your student ID and execute the following search over the **Last 4 hours**:

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



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

    host: `engdev##`

    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 a few 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 a few minutes, review the Troubleshooting Suggestions section.

Task 3: 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 so that it uses an explicit value for **host** and routes the data to the **`sales`** index. You will then re-deploy the updated input to the forwarder. After the updated input is deployed to the forwarder, any new data is sent to sales index, but the data that was indexed into the test index 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.

19. 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
```

20. 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 of the app on the client. The **`reload deploy-server`** command causes the deployment server to re-cache the deployable apps and updates the checksum values of 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.

21. To re-deploy the new **`inputs.conf`** settings, run this command (Splunk may prompt you for the **`admin`** username and password which is **admin** and **your assigned password**.):

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

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

22. Remote SSH into your UF2 (**`10.0.0.100`**) and verify the update has been deployed.

```
cat ~/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)
```

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

23. Trigger the re-indexing of the data in the **sales** index by resetting the monitor checkpoints on the forwarder. Although the supported method is to use **btprobe** to reset each monitored input, this is a test system, so we can use the simple (but also dangerous) methods of either removing the fishbucket folder, or using the **splunk clean evendata** command. We will show all methods in the step and notes below, however only one of these methods needs to be used in the lab.

    The following commands stop splunk, remove the **fishbucket** directory that stores all the fishbucket related files, and then starts splunk. Note that this affects all monitored inputs, however because we are on a test system where we want to reset all checkpoints, this is not a concern.

```
~/splunkforwarder/bin/splunk stop

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

~/splunkforwarder/bin/splunk start

exit
```

NOTE: Instead of removing the **fishbucket** folder, the supported method of resetting monitored file inputs individually on a "production" system is by using the **btprobe** command:

```
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: Another method for resetting all monitored inputs is running the **splunk clean eventdata -index _thefishbucket** command and restarting Splunk. This command should be used with caution, as typos or running on incorrect systems can have disastrous consequences: Running **splunk clean** without the **-index** option will remove ***all*** indexes from that Splunk instance. In this lab, instead of removing the **fishbucket** folder or running the **btprobe** commands, you could instead run:

```
cd ~/splunkforwarder/bin
./splunk stop

./splunk clean eventdata –index _thefishbucket

./splunk start
exit
```

Check Your Work

Task 3: Verify your forwarder is sending the events to the indexer.

24. From your search head, replace the ## with your student ID and execute the following search over the **Last 4 hours**:

    ```
    index=sales host=www-##
    ```

    Eventually, you should see one sourcetype and three sources in your search results. It may take a few 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:

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

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

   2nd phase deployment:

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

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

   ```
   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 6 Lab Exercise – Network Input and Deploy a Remote Scripted Input

Description

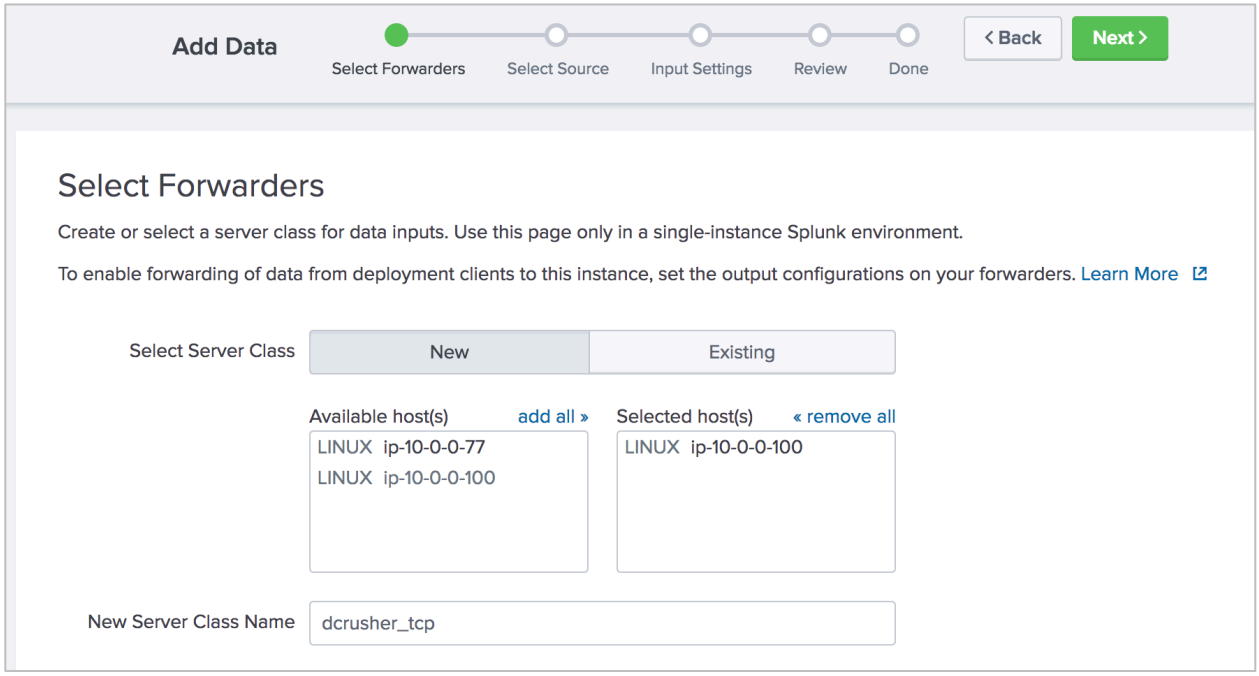
Your instructor has configured a source to send TCP traffic to your UF2 (**`10.0.0.100`**). In the first part of this exercise, you will deploy a network input to the UF2 (**`10.0.0.100`**) which will only receive events from a known host and forward that data to the indexers.

Steps

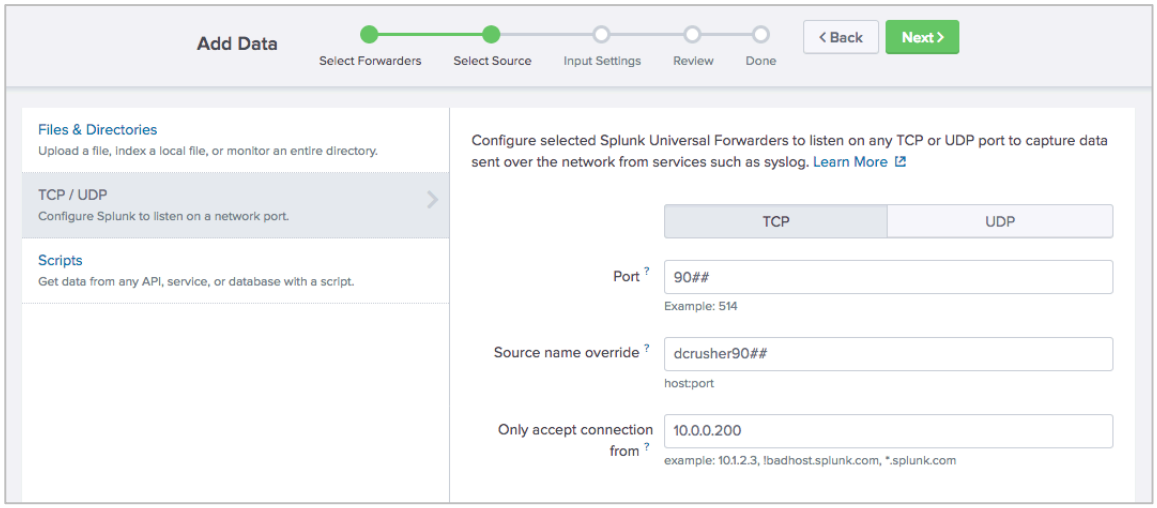
Task 1: Add a forward network input and deploy it to UF2 (10.0.0.100).

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

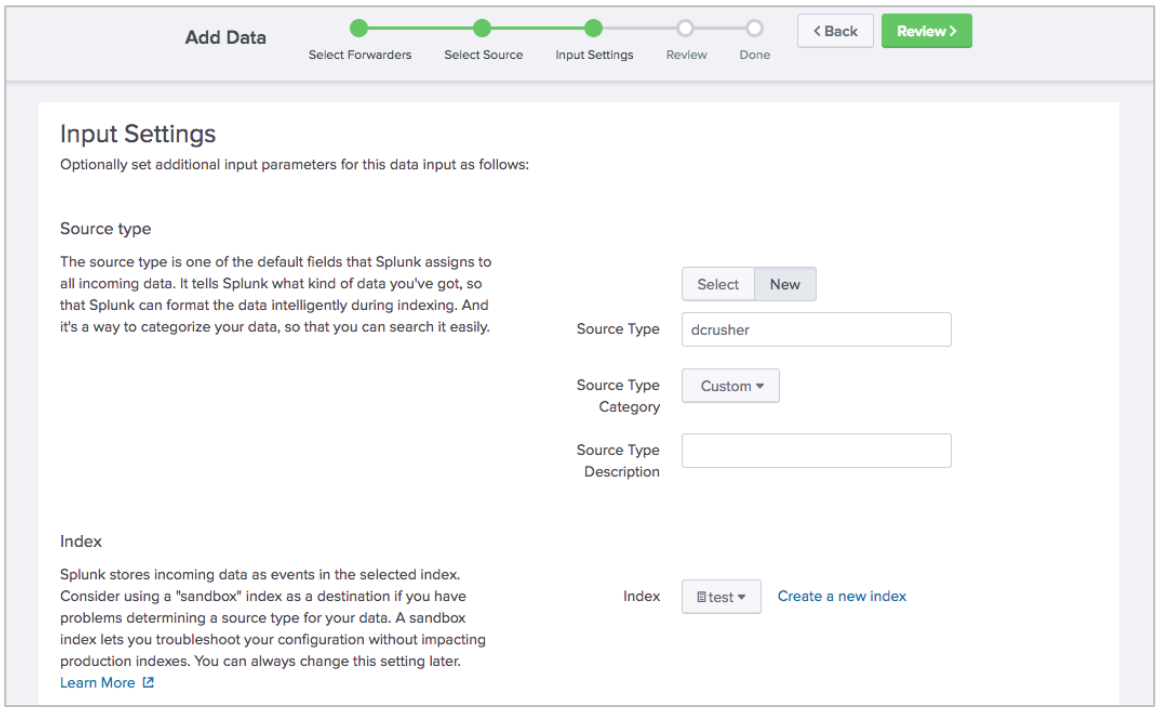
1. On the deployment server, click **Settings > Add Data > Forward**.
2. On the **Select Forwarders** step, configure the form as follows:
   - Select Server Class: **`New`**
   - Selected host(s): **`LINUX ip-10-0-0-100`**
   - New Server Class Name: **`dcrusher_tcp`**



3. On the **Select Source** step, click **TCP / UDP** and configure the form as follows (replace ## with your student ID):
   - 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. From your search head, replace the ## with your student ID and execute the following search over the **Last 15 minutes**:

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

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. You may need to wait a few moments to see results. 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:

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

```
SPLUNK_HOME/etc/deployment-apps/_server_app_dcrusher_tcp/local/inputs.conf
```

10. Edit and save the input stanza as follows:

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

11. To re-deploy the modified input, run:

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

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

Check Your Work (for Network Input)

Task 3: Verify the forwarded TCP input events.

12. From the search head, replace the ## with your student ID and execute the following search over the **Last 15 minutes**:

    ```
    index=itops source=dcrusher90##
    ```

    You should see the following field values:

    host = dcrusher_devserver

    source = dcrusher90##

    sourcetype = dcrusher

Troubleshooting Suggestions (for Network Input)

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.

   ```
   cat ~/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.

Continue on to the next part of lab 6: Deploy a Remote Scripted Input

Description

The Linux **vmstat** command is a useful tool for gathering a snapshot of system 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

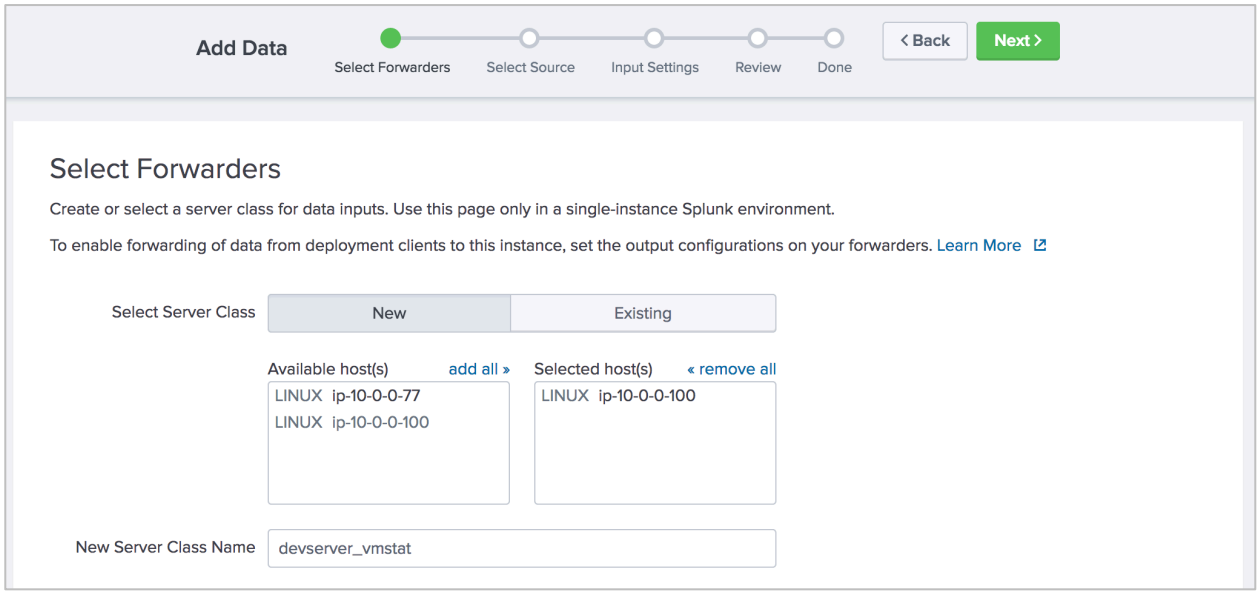
Task 4: Add a scripted input on your deployment server and deploy it to the forwarder #2.

1. From the deployment server's filesystem, copy the **`/opt/scripts/myvmstat.sh`** file to the **`SPLUNK_HOME/bin/scripts`** folder.

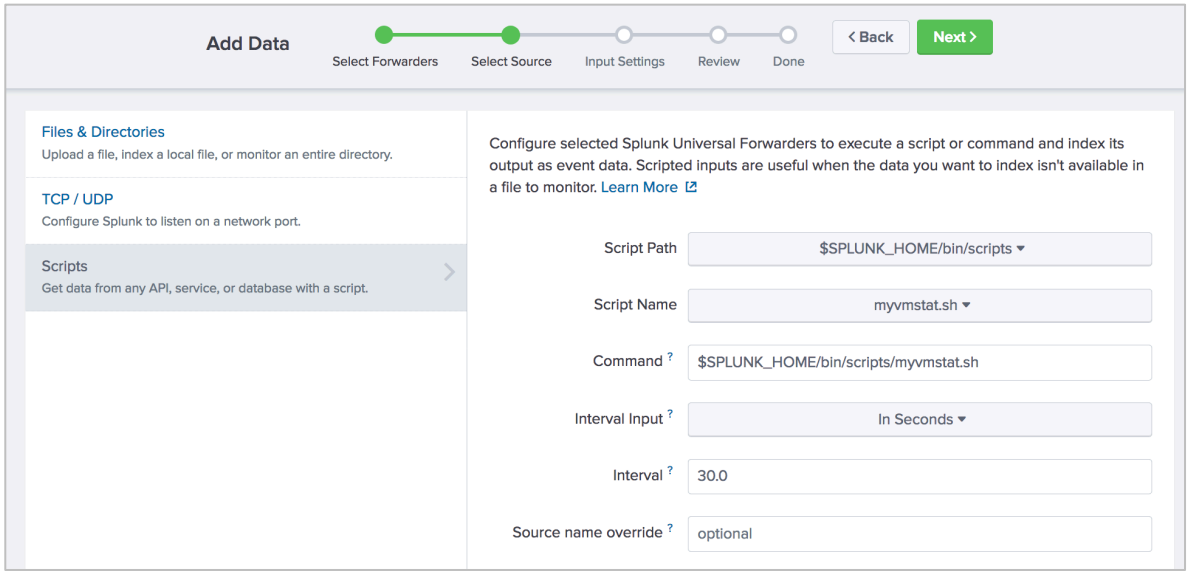
```
cp /opt/scripts/myvmstat.sh /opt/splunk/bin/scripts
```

```
copy \opt\scripts\myvmstat.sh "C:\Program Files\Splunk\bin\scripts\"
```

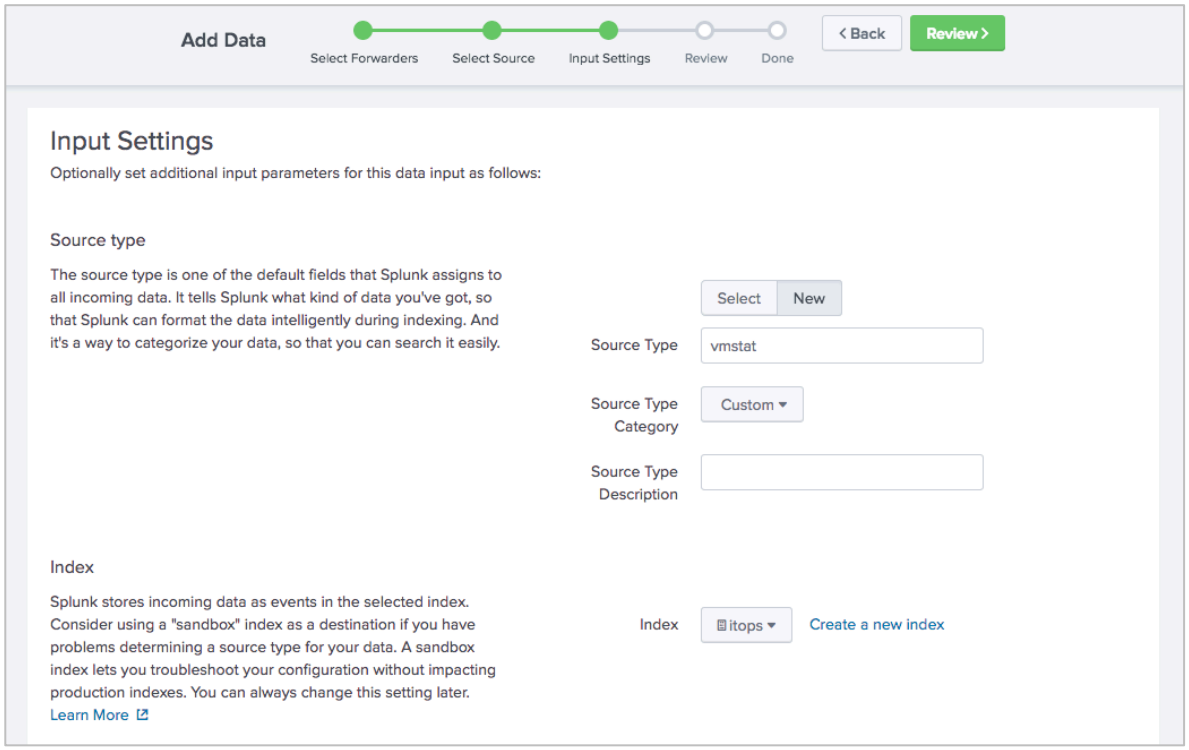
2. From the deployment server, click **Settings > Add Data > Forward**.
3. On the **Select Forwarders** step, configure the form as follows:
   - Select Server Class: **`New`**
   - Selected host(s): **`LINUX ip-10-0-0-100`**
   - New Server Class Name: **`devserver_vmstat`**



4. 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`**
   - Interval: **`30`**



5. For the **Input Settings**, select **New**, enter **Source type** as **`vmstat`** and select **`itops`** for the **Index**, and click **Review**:



6. 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`

7. Click **Submit**.
8. For **Windows Students Only:** Remote SSH to forwarder #2 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 (for Deploy a Remote Scripted Input)

Task 5: Verify the output of your scripted input.

9. From the search head, replace the ## with your student ID and execute the following search over the **Last 15 minutes**.

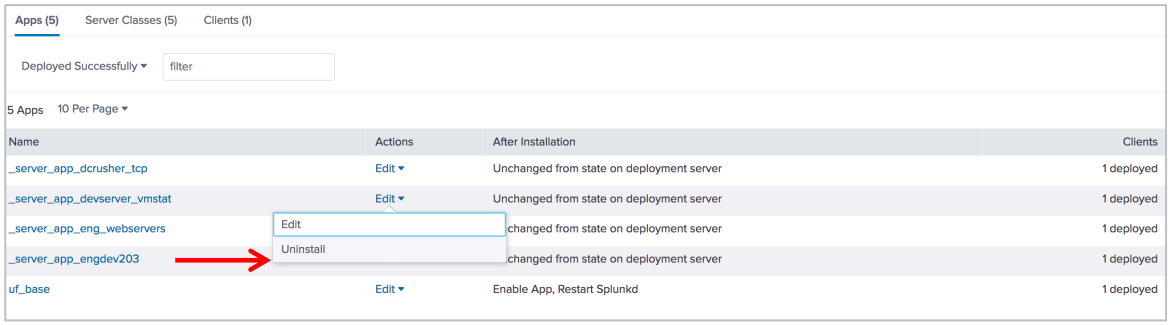
```
index=itops sourcetype=vmstat host=engdev2##
```

You may need to wait a few moments to see results. When you do, do not navigate away from these search results.

Task 6: 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.

10. On the deployment server, navigate to **Settings** > **Forwarder management** and click the **Apps** tab.
11. For the **app** `_server_app_devserver_vmstat`, click **Edit > Uninstall > Uninstall**.



12. Switch back to the search head window/tab and change the time range of the search to: **REAL-TIME > 1 minute window**

13. Wait until the event count drops to **0** (0 of X events matched) and then stop (click ■) the real-time search.

Troubleshooting Suggestions (for Deploy a Remote Scripted Input)

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.

splunk>

Module 7 Lab Exercise – HTTP Event Collector

Description

In this lab exercise, you enable and configure the HTTP event collector 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 HTTP Event Collector Receiver (deployment server).

1. On the deployment server, navigate to **Settings** > **Data inputs**.
2. From Local inputs, click **HTTP Event Collector**.
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`**	*(Leave the box unchecked)*
Enable SSL	**`off`**	*(Uncheck the box)*
HTTP Port Number	**`8088`**	

5. Click **Save**.
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`**
8. From the **Output Group (optional)**, notice that **None** is selected in the dropdown menu and click **Next**.
9. On the **Input Settings** page, set the values to the following:

Source type	**`Automatic`**
App Context	**`Search & Reporting (search)`**
Select Allowed Indexes	Add **`itops`** and **`test`** to the **Selected item(s)**
Default Index	**`test`**

10. 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**

11. 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.

12. Copy the **Token Value** and save it to a text document.

Check Your Work

Task 2: Send test events to your indexer.

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.

13. From your deployment server, remote SSH to your forwarder #1 (**10.0.0.50**).

14. Execute the following commands to set environment variables for the HEC events:

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

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"}'
```

    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
    ```

    Each successful run indexes 10 events. You should see the following field values:

    host = **10.0.0.2##:8088**

    source = **http:iot_sensors**

    sourcetype = **json_no_timestamp**

17. 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. **Important:** The message must begin with your student ID in order to validate the data. (Note: If you have exited the terminal window since running the **export** commands in step #14, ensure you run those commands again prior to running this script.)

    ```
    /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
    {"text":"Success","code":0}
    ...
    ```

    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"'"}}'
    ```

18. Close your remote SSH session.

    ```
    exit
    ```

19. 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 **deploymentclient.conf** file.

```
./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 = ip-10-0-0-2##
/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 8 Lab Exercise – Fine-tuning Inputs

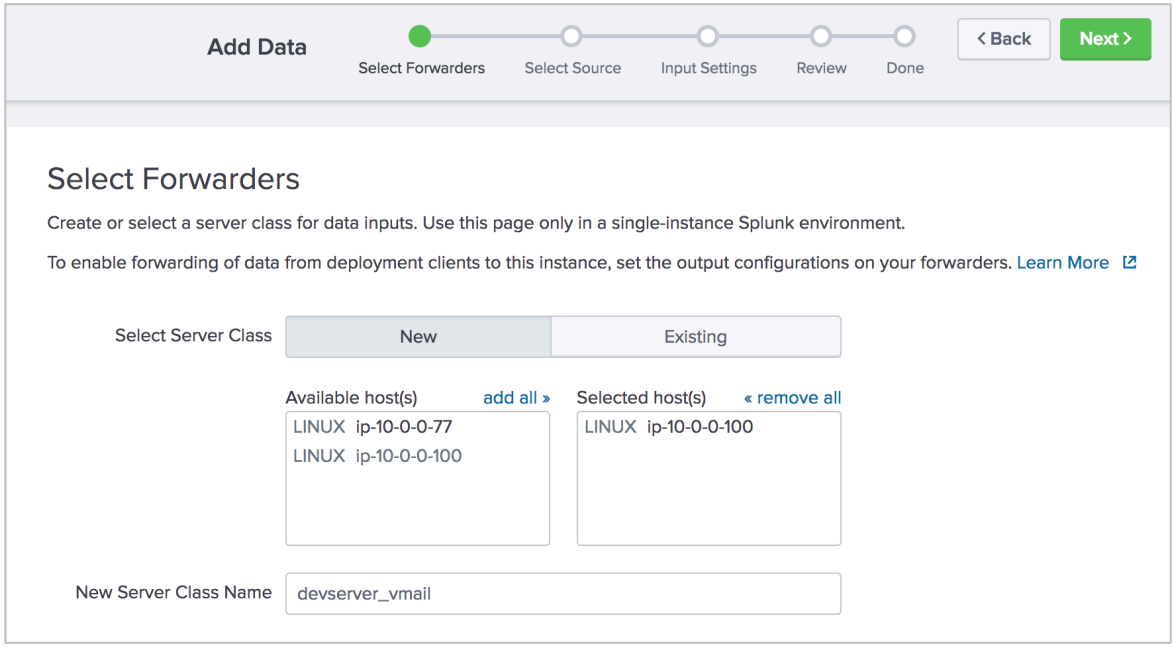
Description

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.

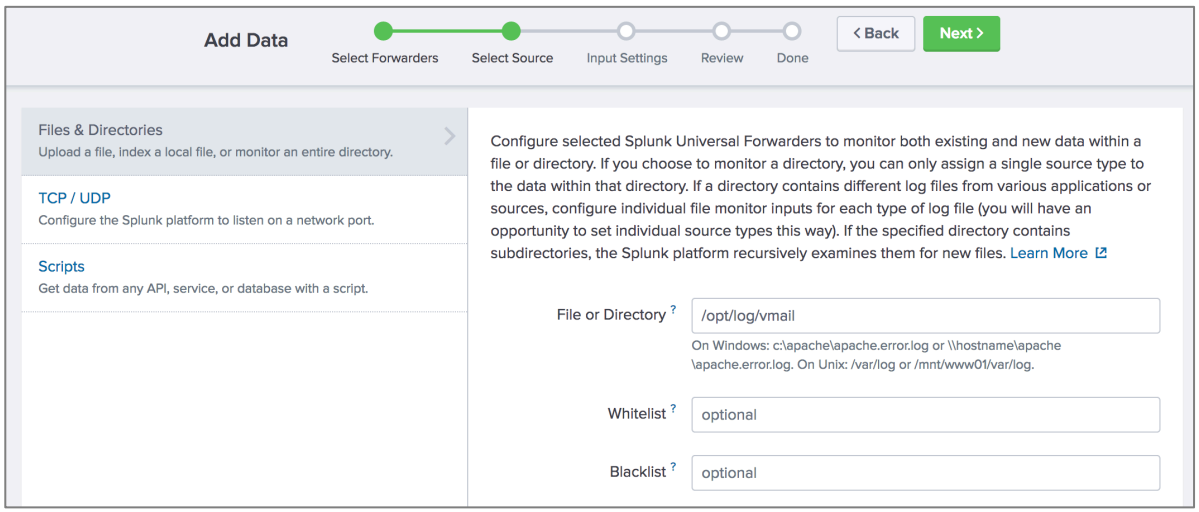
Steps

Task 1: Add a remote test directory monitor input to sample the auto-sourcetype behavior.

1. From your deployment server, click **Settings > Add Data > Forward**.
2. On the **Select Forwarders** step, configure the form as follows:
   - 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 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`**
Whitelist	**`N/A`**
Blacklist	**`N/A`**
Source Type	**`Automatic`**
Index	**`test`**

6. From your search head, execute the following search over the **Last 30 days**, replacing ## 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/iisvmail1.log`** **`/opt/log/vmail/iisvmail2.log`** **`/opt/log/vmail/iisvmail3.log`** **`/opt/log/vmail/iisvmail4.log`**
sourcetype:	**`iis_vmail-2`** and **`iis_vmail`**

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 second forwarder. 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. 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: (Note that 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. Remote SSH into UF2 (**10.0.0.100**) and 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
```

15. To trigger the re-indexing of the same sources on the forwarder, the monitor checkpoints must be reset on UF2 (**10.0.0.100**).

    To clear the individual checkpoints for two of the **iis_vmail** logs, run 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

exit
```

Check Your Work

Task 3: Verify the source type.

16. From the search head, execute the following search over the **Last 30 days**, replacing ## with your student ID:

    ```
    index=itops source=*vmail* host=engdev2## | stats count by source, sourcetype
    ```

17. Confirm that **iis_vmail3.log** is now using the overridden sourcetype **acme_voip**, while the other sources are still using the automatic sourcetype values of **iis_vmail** and/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
    ```

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

splunk>

Module 9 Lab Exercise – 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`**) 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. This is a typical Splunk Cloud configuration. For Splunk on-prem deployments, the **`props.conf`** is deployed to the indexers.

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, click **Settings > Add Data > Monitor**.
2. On the **Select Source** step, click **Files & Directories**.
3. Click **Browse** to navigate and select one of the crash log files (do not select file **dreamcrusher.xml**):

   `/opt/log/crashlog/crash-[DATE].log`

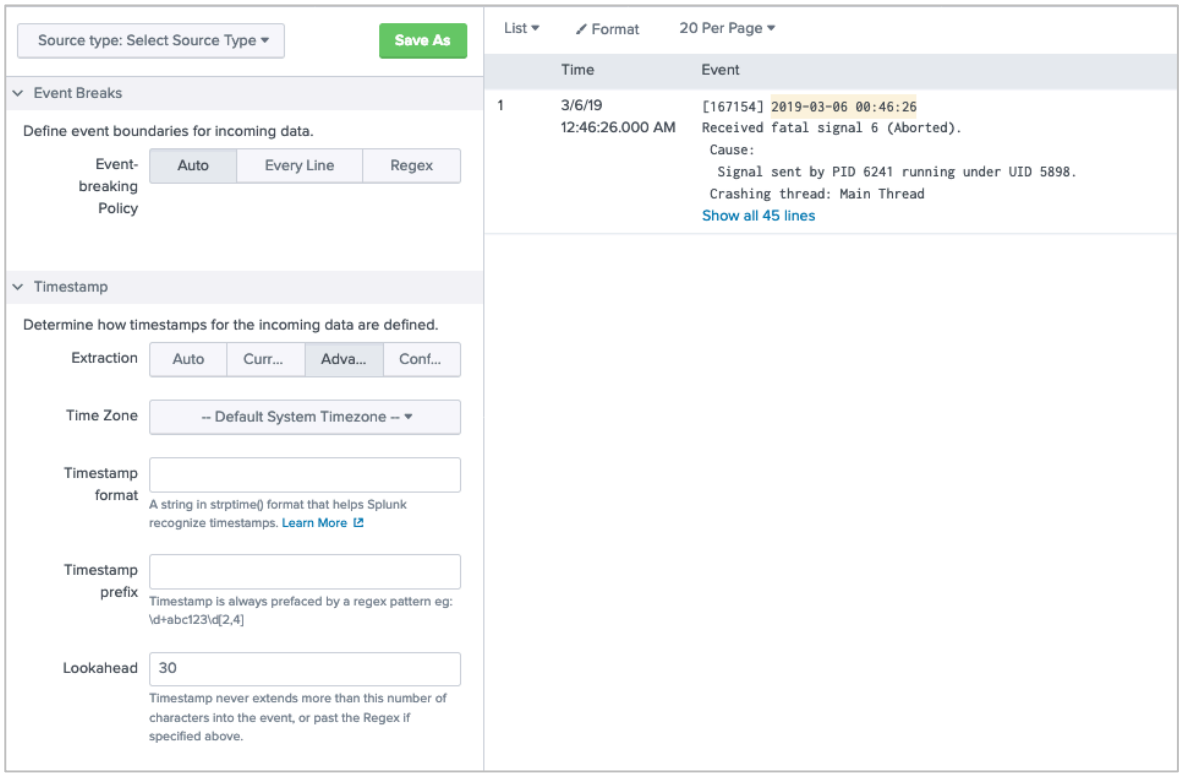
   `C:\opt\log\crashlog\crash-[DATE].log`

4. Verify **Continuously Monitor** is selected and click **Next**.

   On the **Set Source Type** step, note that the ***data preview*** panel displays two events.

5. To have Splunk treat this as a single event using only the timestamp on the first line, click **Timestamp** > **Advanced…**.

6. Change the **Lookahead** value to `30` and press **Tab**.
   After the adjustment, the data preview panel should now display only one event.



7. 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`

8. Click **Save**.
9. Expand the **Advanced** section on the left and click **Copy to clipboard**.
10. Review the `props.conf` attributes produced by your customizations, then click **Cancel**.
11. Click **Next** to proceed to **Input Settings**.
12. On the **Input Settings** step, make sure **App Context** is set to **Search & Reporting (search)** and select `test` for the Index.
13. 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`

14. 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.

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.

15. 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.

16. Launch the **Add Data Wizard** and add a new **Monitor** input. Select **Files & Directories**, and set the **File or Directory** to the full path to the **dreamcrusher.xml** file, then click **Next**.

17. 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

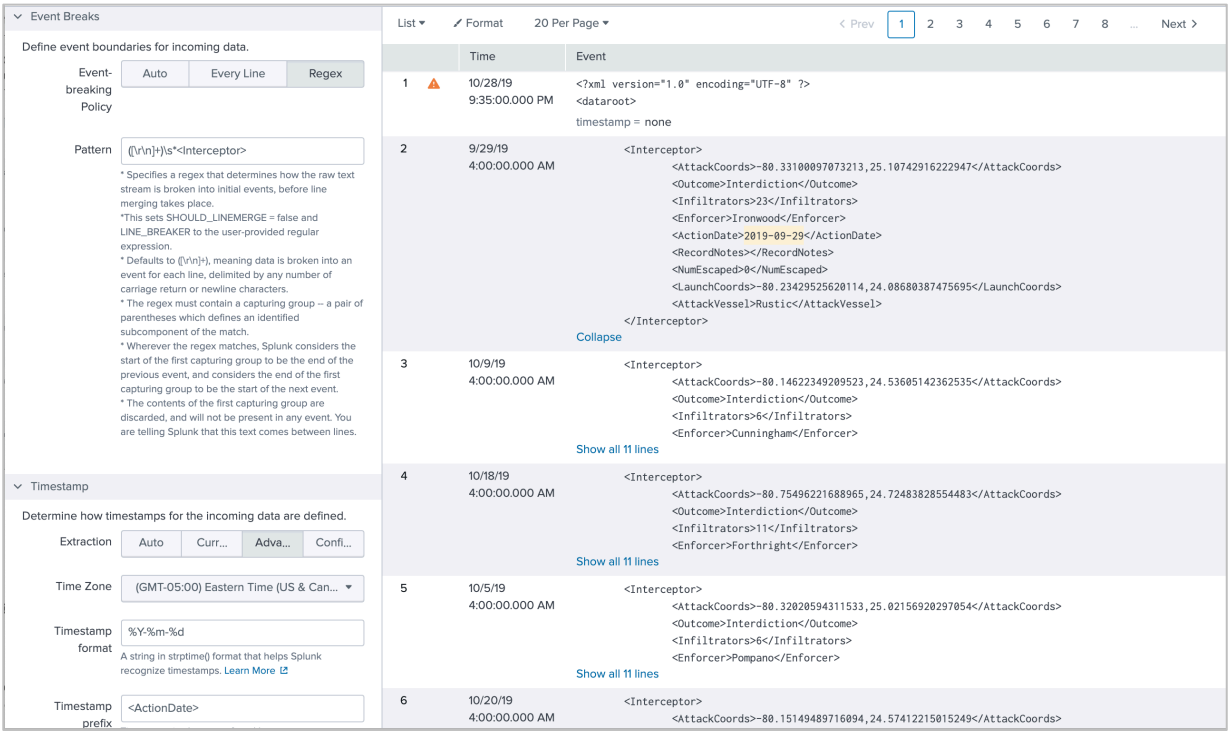
18. Configure the event breaking and the timestamp extraction as follows:

    - Expand the **Event Breaks** section, click **Regex...** and for **Pattern,** type: **([\r\n]+)\s*<Interceptor>**
    - Press **Tab**. (You should see the XML data placed in proper multi-line events.)
    - To see the **<ActionDate>** tag (timestamp) of the second event, click **Show all ## lines**.
    - Expand 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>**

    - Press **Tab**. (You should see the dates updated correctly for these events.)

NOTE: This timestamp extraction will not be applied to the XML root element. You can safely ignore the warning icon on the first event. All subsequent events should no longer display a warning at this point. The displayed timestamps should each have the correct date.



19. Click **Save As** to save the source type configuration as follows:

Name:	`dcrusher_attacks`
Description:	`Dream Crusher user interactions`
Category:	`Application`
App:	`Search & Reporting`

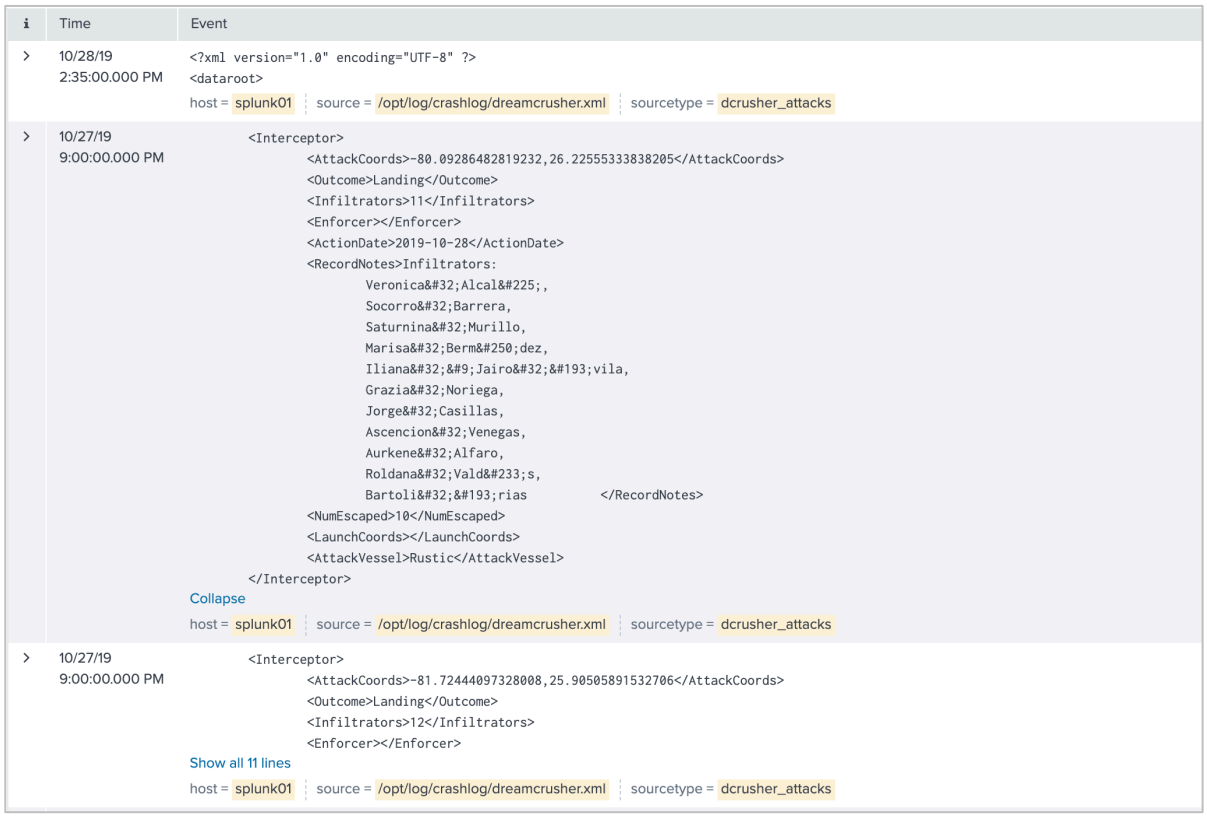
20. Click **Save**, then **Next**.
21. On the **Input Settings** step, make sure the **App Context** is set to **Search & Reporting (search)** and select the `test` index.
22. Verify the **Review** page matches the following:

Input Type	`File Monitor`
Source Path	`/opt/log/crashlog/dreamcrusher.xml` `C:\opt\log\crashlog\dreamcrusher.xml`
Continuously Monitor	`Yes`
Source Type	`dcrusher_attacks`
App Context	`search`
Host	`splunk##`
Index:	`test`

23. Click **Submit**.
24. Click **Start Searching**.

    If each event starts with **<Interceptor>**, displays the correct timestamp, and the sourcetype is set to **dcrusher_attacks**, continue with the next task. Ignore the XML header event containing **<?xml version...**

    If not, consult the **Troubleshooting Suggestions** and repeat the task.



Task 3: Prepare the props.conf file on the deployment/test server.

25. On the deployment server, copy the contents of the **props.conf** file to the **hf_base** directory.

```
cp -r /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"
```

26. 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
```

27. Remote SSH to splunkHF## (**10.0.0.77**) and make sure the new **[dc_mem_crash]** and **[dcrusher_attacks]** stanzas appear in the deployed **props.conf** file:

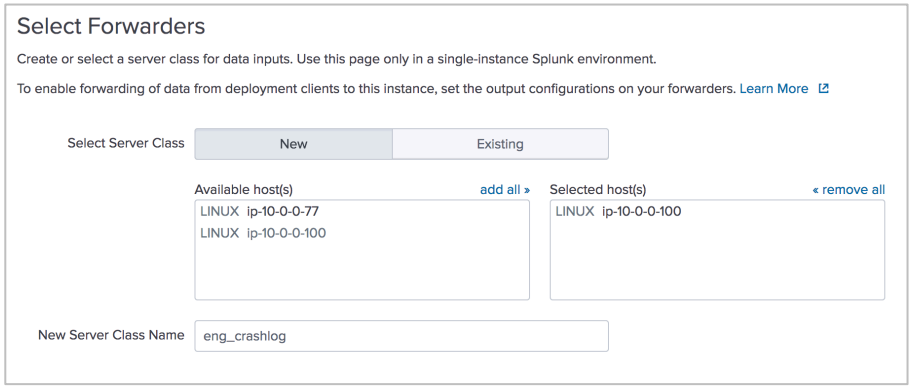
    Your **dc_mem_crash** and **dcrusher_attacks** stanzas should match the output shown below:

```
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

exit
```

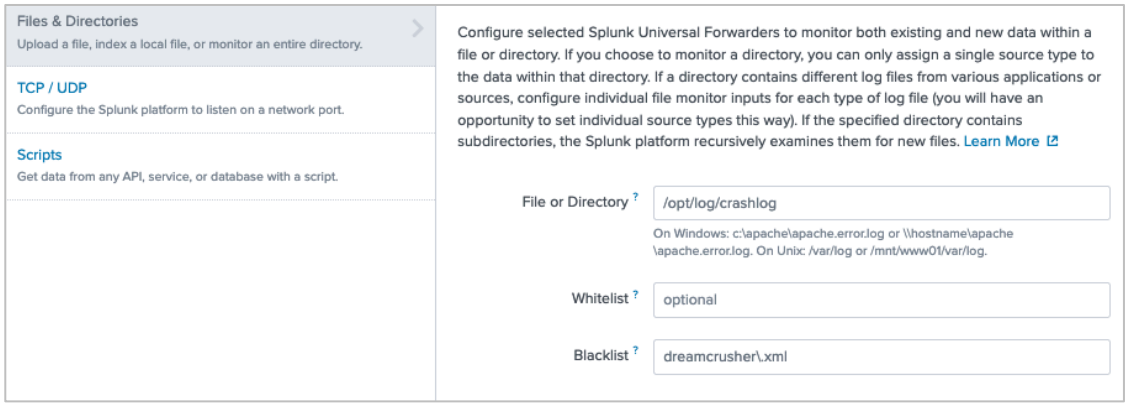
Task 4: Deploy a directory monitor to UF2 to index the crash logs into the test index.

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.

28. From the deployment server, click **Settings > Add Data > Forward**.
29. On the **Select Forwarders** step, configure the form as follows:
    - Select Server Class: **New**
    - Selected host(s): **LINUX ip-10-0-0-100**
    - New Server Class Name: **eng_crashlog**



30. 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`**
    - Blacklist: **`dreamcrusher\.xml`**



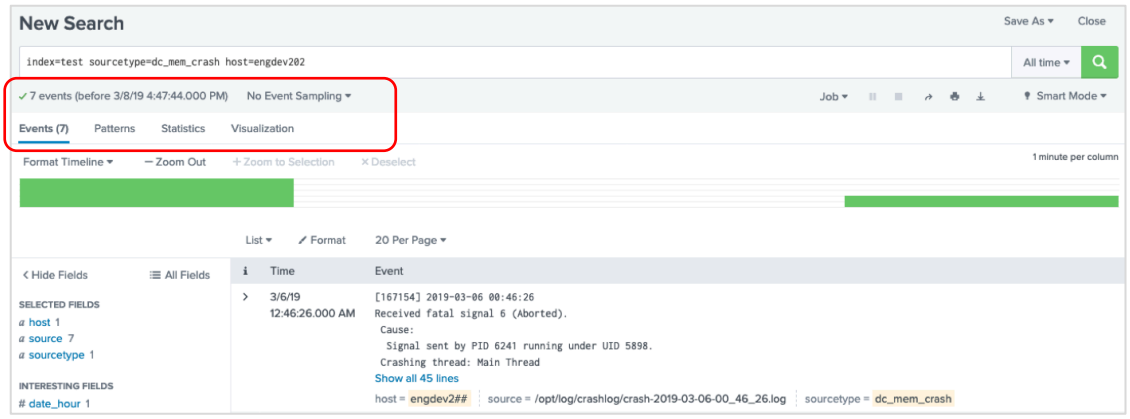
31. For the **Input Settings**, for the Source type click on **Select** and select **Application > `dc_mem_crash`** sourcetype defined earlier, and the **`test`** index.

32. 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`**
Whitelist	**`N/A`**
Blacklist	**`dreamcrusher\.xml`**
Source Type	**`dc_mem_crash`**
Index	**`test`**

33. From the search head, execute the following search over **All Time**, replacing ## with your student ID:

    `index=test sourcetype=dc_mem_crash host=engdev2##`



    You should see that a total of 7 events and 7 sources are being parsed by the definitions defined in the deployed **props.conf** file. Note that it may take a few minutes before the results show. If you see 14 events, stop and 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.

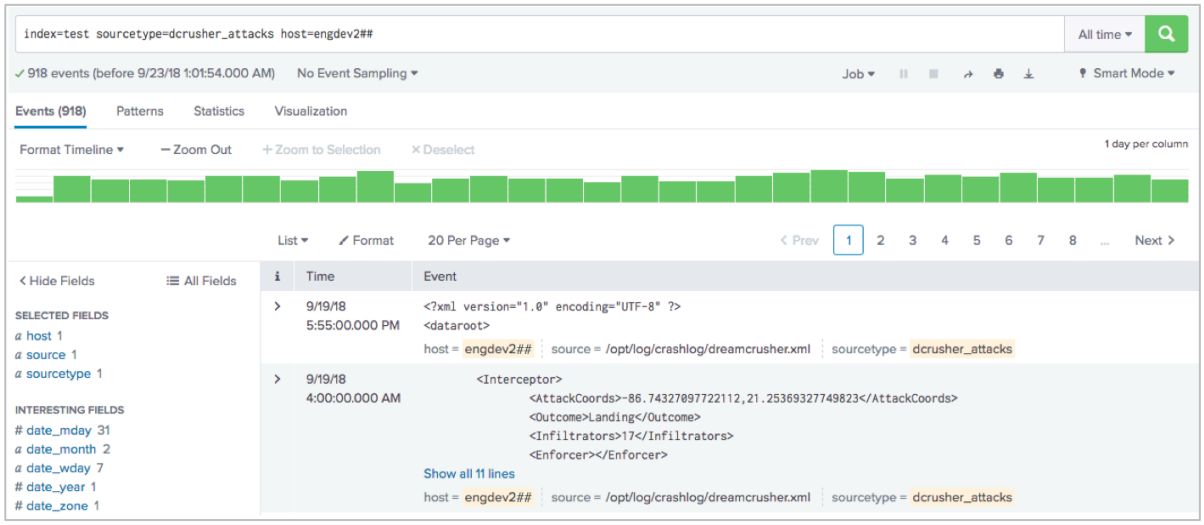
In this task, you add a Forward input to monitor **`dreamcrusher.xml`** on UF2. The XML file is forwarded to your heavy forwarder for line breaking and timestamp extraction. The parsed events are then forwarded to the indexers.

34. From the deployment server, launch the **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`**
35. 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`**
Whitelist	**`N/A`**
Blacklist	**`N/A`**
Source Type	**`dcrusher_attacks`**
Index	**`test`**

36. From the search head, execute the following search over **All Time**, replacing the ## with your student ID (it can take a few minutes for the data to display on the search head):

    ```
    index=test sourcetype=dcrusher_attacks host=engdev2##
    ```



    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:

    - 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,

```
cat /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
```

```
cat /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 three lines but it should be entered on a single line.
Replace **<xxxx...>** with the actual name.
If you still don't get results, ask your instructor for help.

splunk>

Module 10 Lab Exercise – Manipulating Data

Description

In this lab exercise, you ingest two log files on UF2 and perform the following data manipulation tasks on the heavy forwarder:

- Evaluate data forwarded and mask account codes before data is transmitted to the indexers.
- Configure data from the same source to be indexed into different indexes while filtering out (discarding) unwanted data, all based on a keyword match.

Steps

Task 1A: Create a data masking transformation.

1. On the deployment/test server, open the sample file in a text editor.

   `/opt/log/ecommsv1/sales_entries.log`

   `C:\opt\log\ecommsv1\sales_entries.log`

NOTE: The **AcctCode** field values contain sensitive account numbers; these numbers should be masked for privacy reasons.

2. 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`

3. Add the following stanza to mask the last four digits of the **AcctCode** field values:

```
[mask-acctcode]
REGEX = (.*AcctCode=\d{4}).*
DEST_KEY = _raw
FORMAT = $1-XXXX
```

4. 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`

5. Append the following stanza to invoke the **acctmasking** transformations for the **sales_entries** source type:

```
[sales_entries]
TRANSFORMS-acctmasking = mask-acctcode
```

6. Restart the deployment server.

```
/opt/splunk/bin/splunk restart
```

```
C:\Program Files\Splunk\bin\splunk restart
```

Task 2A: Add a local file monitor input to test the transformation.

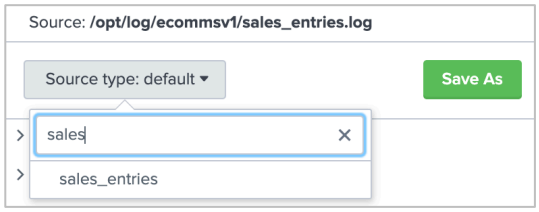
7. Log back into the deployment/test server and launch the **Add Data** wizard.
8. Add a **Monitor** (local) input and, on the **Select Source** step, select **Files & Directories**.
9. Click **Browse** and select the following file:

```
/opt/log/ecommsv1/sales_entries.log
```

```
C:\opt\log\ecommsv1\sales_entries.log
```

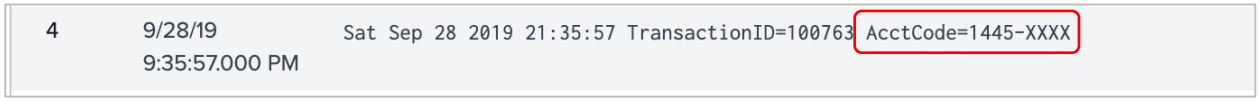
10. Verify **Continuously Monitor** is selected and click **Next**.
11. Click **Source type: default** and type "**sales**".

    The **sales_entries** source type should display.



12. Select **sales_entries** for the source type.

    After selecting the **sales_entries** sourcetype, the **AcctCode** values in the data preview pane should be masked.



    IMPORTANT: If the **AcctCode** 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 1A, Steps 2-5) and repeat this step.

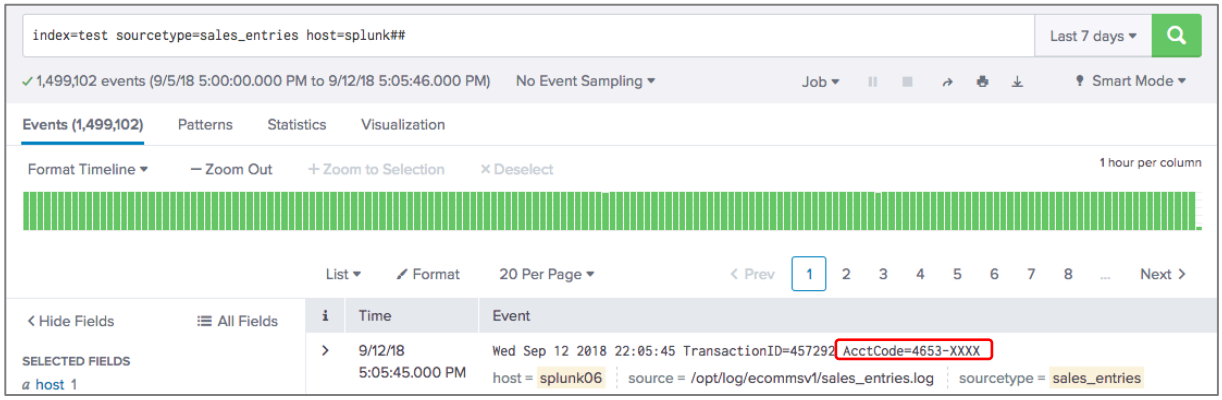
13. Click **Next** and select **test** for the index setting.
14. Click **Review** and verify that your input matches the following before clicking **Submit**.

Input Type	**File Monitor**
Source Path	**/opt/log/ecommsv1/sales_entries.log** **C:\opt\log\ecommsv1\sales_entries.log**
Continously Monitor	**Yes**
Source Type	**sales_entries**
App Context	**search**
Host	**splunk##**
Index	**test**

15. From the deployment/test server, execute the following search over the **Last 7 days**, replacing the ## with your student ID:

    ```
    index=test sourcetype=sales_entries host=splunk##
    ```

    You should see events with the last four digits of the **AcctCode** field masked.



Task 3A: Copy the props and transforms file definitions to the hf_base app and deploy them to the HF.

16. Copy the contents of the **props.conf** file to the **hf_base** directory.

```
cp -r /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"
```

17. Copy the contents of the **transforms.conf** file to the **hf_base** directory.

```
cp -r /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"
```

18. 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
```

19. Remote SSH to HF## (**10.0.0.77**) and confirm the new **[sales_entries]** stanza below appears with the other stanzas in the deployed **props.conf** file.

```
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/Chicago
category = Application
description = Dream Crusher user interactions
disabled = false
pulldown_type = true

[sales_entries]
TRANSFORMS-acctmasking = mask-acctcode
```

20. Confirm the new stanza appears in the deployed **transforms.conf** file.

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

[mask-acctcode]
REGEX = (.*AcctCode=\d{4}).*
DEST_KEY = _raw
FORMAT = $1-XXXX

exit
```

Task 4A: Deploy a file monitor to UF2 to transmit the sales_entries.log data.

21. Launch the **Add Data** wizard and add a **Forward** input to monitor **sales_entries.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_saleslog**
- 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**, type "**sales**" and select **sales_entries**
  Index **itops**

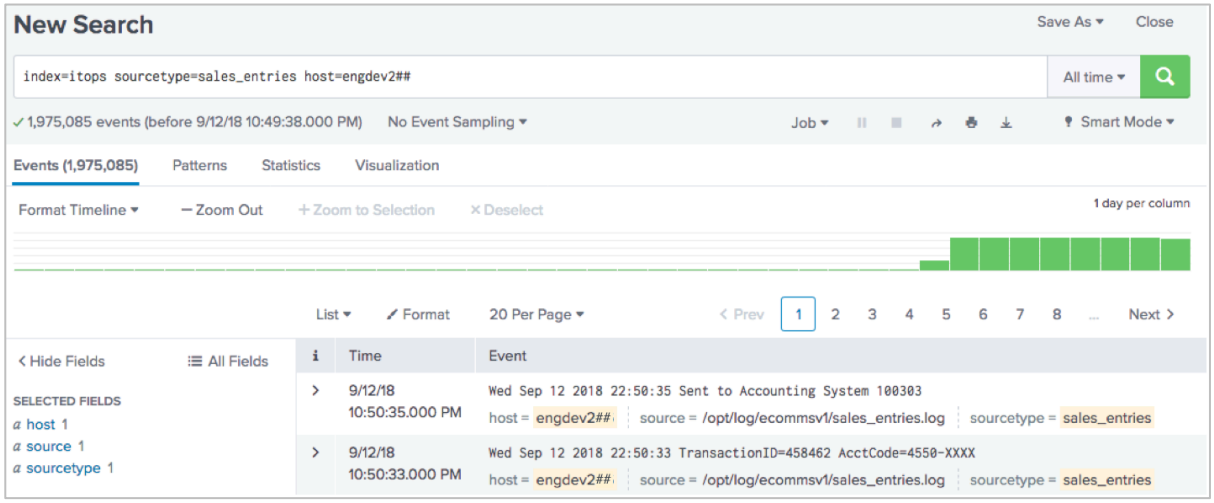
22. Verify the **Review** page matches the following before clicking **Submit**:

Server Class Name	`eng_saleslog`
List of Forwarders	`LINUX|ip-10-0-0-100`
Input Type	`File Monitor`
Source Path	`/opt/log/ecommsv1/sales_entries.log`
Whitelist	`N/A`
Blacklist	`N/A`
Source Type	`sales_entries`
Index	`itops`

23. From your search head, execute the following search over **All Time**, replacing the ## with your student ID:

```
index=itops sourcetype=sales_entries host=engdev2##
```

You should see events from UF2 with the last four digits of the **`AcctCode`** field masked. 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.

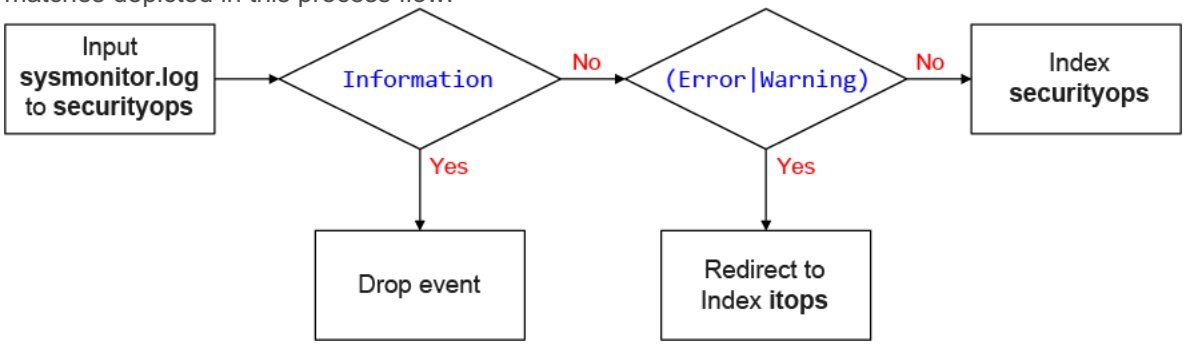
1. Verify the syntax and spelling in all configurations and searches.

The next part of lab 10: Data Filtering and Redirecting Configuration Steps, is optional.

splunk>

Module 10 Optional Lab Exercise – Data Filtering and Redirecting Configuration Steps

Now you will configure the heavy forwarder to perform 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 Splunk to either drop or redirect individual events based on REGEX pattern matches depicted in this process flow:



Task 1B: 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.

1. On the deployment/test server, open the sample file in a text editor.

   `/opt/log/adldapsv1/sysmonitor.log`

   `C:\opt\log\adldapsv1\sysmonitor.log`

   Review the event data. Each event contains one of five keywords: **`Error`**, **`FailureAudit`**, **`Information`**, **`SuccessAudit`**, or **`Warning`**. Close the file when done.

2. 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`

3. 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
```

4. 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
```

5. Append the following stanza to invoke the filtering and routing transformations for the **win_audits** sourcetype:

```
[win_audits]
TRANSFORMS-information = eventsDrop
TRANSFORMS-securityops = eventsRoute
```

6. Restart the deployment server.

```
/opt/splunk/bin/splunk restart
```

```
C:\Program Files\Splunk\bin\splunk restart
```

Task 2B: Add a local file monitor input to test the transformations.

7. Log back into the deployment server and launch the **Add Data** wizard.
8. Add a **Monitor** (local) input and select **Files & Directories**.
9. Click **Browse** and select the following file:

```
/opt/log/adldapsv1/sysmonitor.log
```

```
C:\opt\log\adldapsv1\sysmonitor.log
```

10. Verify **Continuously Monitor** is selected and click **Next**.
11. Click Source type: default and type: "**win**"

    The **win_audits** source type should display.
12. Select **win_audits**.

NOTE:	If **win_audits** is not available, then make sure you have completed Task 1B, Steps 4 and 5 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**.

13. Click **Save As > Save** to save the new source type definitions for the **win_audits** source type.
14. When you are prompted, click **OK**.
15. Click **Next**. From the **Input Settings** step, select **securityops** in the **Index Name** field.

16. Click **Review** and verify that your input matches the following before clicking **Submit**.

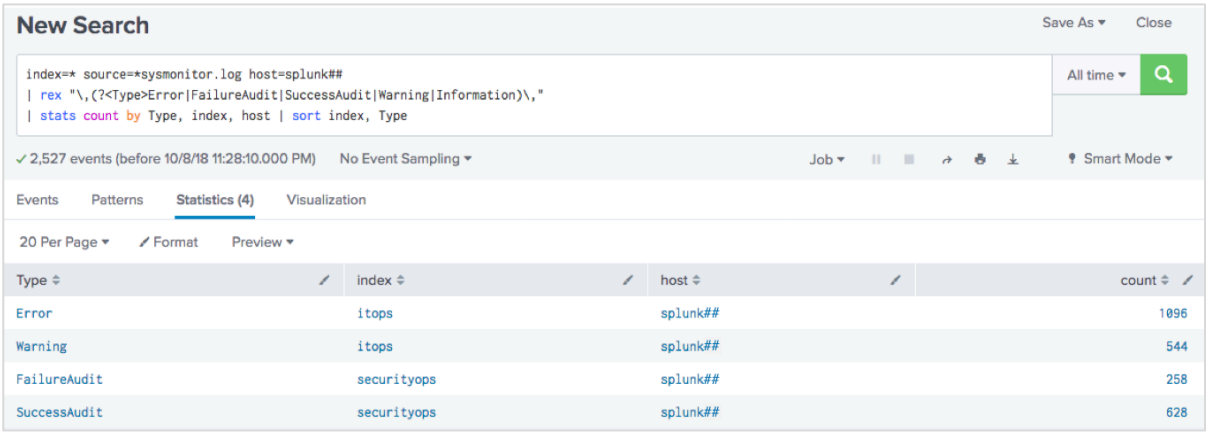
Input Type	`File Monitor`
Source Path	`/opt/log/adldapsv1/sysmonitor.log` `C:\opt\log\adldapsv1\sysmonitor.log`
Continously Monitor	`Yes`
Source Type	`win_audits`
App Context	`search`
Host	`splunk##`
Index	`securityops`

Check Your Work

Task 3B: Make sure events are being filtered and routed properly.

17. From the deployment/test server, execute the following search over **All Time**, replacing the ## with your student ID:

```
index=* source=*sysmonitor.log host=splunk##
| rex "\,(?<Type>Error|FailureAudit|SuccessAudit|Warning|Information)\,"
| stats count by Type, index, host | sort index, Type
```



Type	index	host	count
Error	itops	splunk##	1096
Warning	itops	splunk##	544
FailureAudit	securityops	splunk##	258
SuccessAudit	securityops	splunk##	628

You should not see any “Information” events.

Create a Remote Monitor for the sysmonitor.log

Task 4B: Copy the new props and transforms conf file definitions to the hf_base app and deploy them to the HF.

18. 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"
```

19. 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"
```

20. 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
```

21. Remote SSH to HF## (**10.0.0.77**) and confirm the new **[win_audits]** stanza appears with the other stanzas in the deployed **props.conf** file.

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

...

[sales_entries]
TRANSFORMS-acctmasking = mask-acctcode

[win_audits]
TRANSFORMS-information = eventsDrop
TRANSFORMS-securityops = eventsRoute
DATETIME_CONFIG = CURRENT
LINE_BREAKER = ([\r\n]+)
NO_BINARY_CHECK = true
SHOULD_LINEMERGE = false
disabled = false
pulldown_type = true
```

22. Confirm the new **[eventsDrop]** and **[eventsRoute]** stanzas appear in the deployed **transforms.conf** file.

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

[mask-acctcode]
REGEX = (.*AcctCode=\d{4}).*
DEST_KEY = _raw
FORMAT = $1-XXXX

[eventsDrop]
REGEX = Information
DEST_KEY = queue
FORMAT = nullQueue

[eventsRoute]
REGEX = (Error|Warning)
DEST_KEY = _MetaData:Index
FORMAT = itops
```


Task 5B: Deploy a file monitor to UF2 to transmit the sysmonitor.log data.

24. Launch the **Add Data** wizard and add a **Forward** input to monitor **`sysmonitor.log`** on UF2. Send the data to the **`securityops`** index.

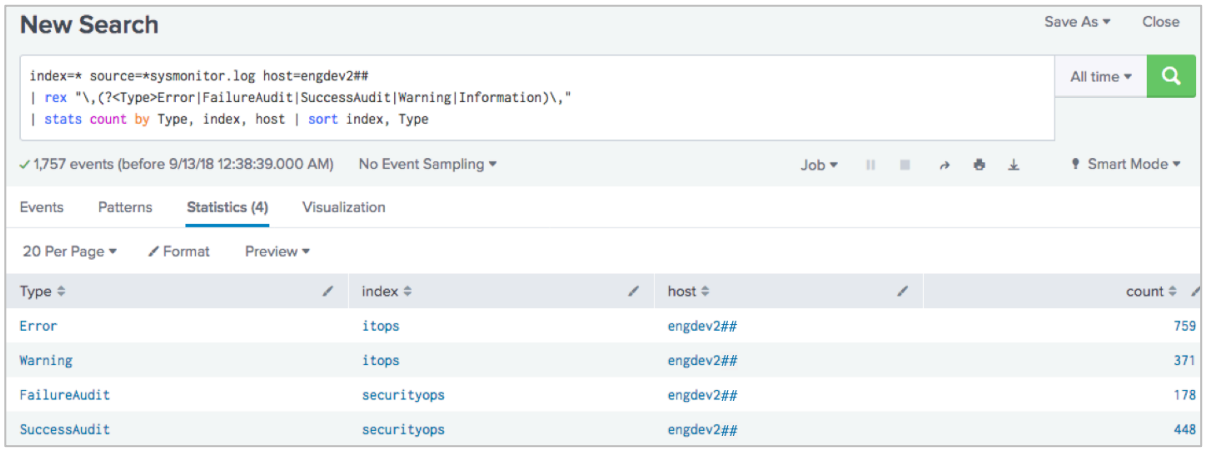
    - On the **Select Forwarders** step:
      Selected Server Class — **New**
      Selected host(s) — **`LINUX|ip-10-0-0-100`**
      New Server Class Name — **`eng_sysmonitor`**
    - On the **Select Source** step, select **Files & Directories**:
      File or Directory — **`/opt/log/adldapsv1/sysmonitor.log`**
    - On the **Input Settings** step:
      Source type — **Select**, type "**`win`**" and select **`win_audits`**
      Index — **`securityops`**

25. 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`**
Whitelist	**`N/A`**
Blacklist	**`N/A`**
Source Type	**`win_audits`**
Index	**`securityops`**

26. From the search head, execute the following search over **All Time**, replacing the ## with your student ID. (Note: it may take a minute for results to show.)

```
index=* source=*sysmonitor.log host=engdev2##
| rex "\,(?<Type>Error|FailureAudit|SuccessAudit|Warning|Information)\,"
| stats count by Type, index, host | sort index, Type
```



Troubleshooting Suggestions

If your searches are not producing the expected results, check your configurations.

2. Verify the syntax and spelling in all configurations and searches.
3. 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
```

splunk>

Module 11 Lab Exercise – Reassign a Knowledge Object

Description

In this exercise, you will create a knowledge object (a report) and reassign it to another user.

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. Under the **App: Searching and Reporting** dropdown, click **Reports** and select and view the report you just created.

Task 2: Look for orphaned knowledge objects.

7. Click **Settings > All configurations**.
8. Click **Reassign Knowledge Objects**.
9. From the **All | Orphaned** button, click **Orphaned**.
10. Select **Filter by Owner > All**.

    You should see **No knowledge objects found** indicating no orphaned knowledge objects.

Task 3: Assign the report created in Task 1 to emaxwell.

11. From the **All | Orphaned** button, click **All**.
12. In the **Filter by Owner** field, type your student ID and press **Enter**.

    You should see your report listed.
13. Click the **Reassign** button under the **Actions** column.
14. From the **Reassign Entity** dialog box, click the **New Owner** dropdown.
15. Select `(emaxwell)` and click **Save**.

Task 4: Verify you no longer can see the report.

16. Click on **App** dropdown and select **Searching and Reporting**, then click **Reports**.

    You should not see your report.
17. Click **Yours**.

    You should not see your report.

Task 5: Log into the deployment/test server and verify knowledge object assignment.

18. Log out from Spunk Web, and log back into the deployment/test server as **`emaxwell`** (password **`open.sesam3`**)
19. On the left pane titled **Apps**, click on **Search & Reporting**, then click **Reports**.

    You should see your report listed.
20. Click your report to run and test it.

