Splunk4Admins - Forwarder Management

Lab Guide

Overview

This lab guide contains the hands-on exercises for the Splunk4Admins - Forwarder Management workshop. Before proceeding with these exercises, please ensure that you have a copy of the workshop slide deck, which will help to put into context the tasks you are carrying out.

Download the workshop slide deck: https://splk.it/S4A-FM-Attendee

Prerequisites

In order to complete these exercises, you will need your own Splunk instance. Splunk's hands-on workshops are delivered via the Splunk Show portal and you will need a splunk.com account in order to access this.

If you don't already have a Splunk.com account, please create one here before proceeding with the rest of the workshop.

⚠️ Troubleshooting Connectivity

If you experience connectivity issues with accessing either your workshop environment or the event page, please try the following troubleshooting steps. If you still experience issues please reach out to the team running your workshop.

- **Use Google Chrome** (if you're not already)
- If the event page (i.e. *https://show.splunk.com/event/<eventID>*) didn't load when you clicked on the link, try **refreshing the page**
- **Disconnect from VPN** (if you're using one)
- **Clear your browser cache and restart your browser** (if using Google Chrome, go to: Settings > Privacy and security > Clear browsing data)
- **Try using private browsing mode** (e.g. Incognito in Google Chrome) to rule out any cache issues
- **Try using another computer** such as your personal computer - all you need is a web browser! Cloud platforms like AWS can often be blocked on corporate laptops.

Table of Contents

Exercise – Setup the Deployment Server..... 3
Description..... 3
Task 1: Access the Web UI and CLI of your Deployment Server..... 3
Task 2: Retrieve Splunk settings from your deployment server using the CLI..... 5
Task 3: Examine Splunk configuration file documentation and basic .conf files..... 7
Task 4: Create the outputs base app and enable listening on the indexer..... 10
Task 5: Create the deployment client base app..... 13
Task 6: Create the serverclass.conf file..... 15
Task 7: Connect UF1 and HF1 to the Deployment Server for remote management..... 17
Task 8: Create an inputs app and deploy it from the Deployment server..... 18
Extra Credit Task 1: Create a universal forwarder tarball image..... 21
Useful Deployment Server Commands..... 22
From the Deployment Server (DS):..... 22
From the Deployment Client:..... 22
Links..... 23

Exercise – Setup the Deployment Server

Description

SSH to the deployment server and configure the associated configuration files so the deployment server can manage forwarders.

Use case: Splunk admin needs to be able to manage configurations on universal forwarders remotely.

- Understand the requirements for the use case

  Needs to be able to:

  - Configure the deployment server.
  - Create and manage server classes.
  - Deploy configurations (inputs.conf, outputs.conf) to different server classes.
  - Monitor and verify the deployment status.

Task 1: Access the Web UI and CLI of your Deployment Server

1. Navigate to Deployment Server (DS) web UI: **https://{ip-address}:8444**
2. Log in using your assigned username/password (admin/5p1unk.conf).
3. Dismiss any unnecessary informational messages.
4. Click **Settings**.
   a. The full list of options is displayed for this role. You are assigned the admin role with full administrator privileges on this Splunk instance.
5. Connect to the command line of your dedicated Splunk deployment/test server.
   a. If your Splunk lab server is Linux, use one of these two methods:
      i. If your personal computer is running *nix or macOS, start an SSH session to your deployment/test server by opening a terminal window and executing:

         ```
         ssh -p 2222 splunk@{DS-eip}
         ```

         When prompted for the authenticity of the host and the key fingerprint, type “yes.”
   b. OR, if your personal computer is Windows, use an SSH client, such as PuTTy. (PuTTY is a free and reliable SSH client that can be found at https://www.putty.org). To use PuTTy to start an SSH session to your deployment/test server:
      i. Replace **`splunk@{DS-eip}`** with your designated values.

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

iii. Save the session for later re-use.

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

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

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

1. Navigate to the SPLUNK_HOME directory:

```
cd /opt/showlab/s4a-fm-ds/splunk/bin
```

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

```
./splunk status
```

```
root@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin# ./splunk status
splunkd is running (PID: 3221187).
splunk helpers are running (PIDs: 3221188 3221322 3221383 3221437).
root@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin#
```

The output shows the running status and the splunkd process IDs

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

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

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

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

```
./splunk version
```

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin$ ./splunk version
Splunk 9.0.5 (build e9494146ae5c)
```

```
./splunk show web-port
```

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin$ sudo ./splunk show web-port
WARNING: Server Certificate Hostname Validation is disabled. Please see server.conf/[sslConfig]/cliVerifyServerName for details.
Splunk username: admin
Password:
Web port: 444
```

```
./splunk show splunkd-port
```

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin$ sudo ./splunk show splunkd-port
WARNING: Server Certificate Hostname Validation is disabled. Please see server.conf/[sslConfig]/cliVerifyServerName for details.
Splunkd port: 8091
```

```
./splunk show appserver-ports
```

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin$ sudo ./splunk show appserver-ports
WARNING: Server Certificate Hostname Validation is disabled. Please see server.conf/[sslConfig]/cliVerifyServerName for details.
Application server ports on loopback interface: 8066
```

```
./splunk show kvstore-port
```

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/bin$ sudo ./splunk show kvstore-port
WARNING: Server Certificate Hostname Validation is disabled. Please see server.conf/[sslConfig]/cliVerifyServerName for details.
KV Store port: 8192
```

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

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

These messages can be safely ignored in your workshop environment.

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

1. From your terminal window, navigate to the SPLUNK_HOME/etc/system directory:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/system
```

2. View the files in the README directory:

```
ls README/
```

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

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/system$ cd /opt/splunk-ds/splunk/etc/system
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/system$ ls README/
alert_actions.conf.example      eventdiscoverer.conf.spec     multikv.conf.spec               times.conf.example
alert_actions.conf.spec         eventtypes.conf.example       outputs.conf.example            times.conf.spec
app.conf.example                eventtypes.conf.spec          outputs.conf.spec               transactiontypes.conf.example
app.conf.spec                   federated.conf.example        passwords.conf.example          transactiontypes.conf.spec
audit.conf.example              federated.conf.spec           passwords.conf.spec             transforms.conf.example
audit.conf.spec                 fields.conf.example           procmon-filters.conf.example    transforms.conf.spec
authentication.conf.example     fields.conf.spec              procmon-filters.conf.spec       ui-prefs.conf.example
authentication.conf.spec        global-banner.conf.example    props.conf.example              ui-prefs.conf.spec
authorize.conf.example          global-banner.conf.spec       props.conf.spec                 ui-tour.conf.example
authorize.conf.spec             health.conf.example           pubsub.conf.example             ui-tour.conf.spec
bookmarks.conf.example          health.conf.spec              pubsub.conf.spec                user-prefs.conf.example
bookmarks.conf.spec             indexes.conf.example          restmap.conf.example            user-prefs.conf.spec
checklist.conf.spec             indexes.conf.spec             restmap.conf.spec               user-seed.conf.example
collections.conf.example        inputs.conf.example           savedsearches.conf.example      user-seed.conf.spec
collections.conf.spec           inputs.conf.spec              savedsearches.conf.spec         viewstates.conf.example
commands.conf.example           instance.cfg.example          searchbnf.conf.example          viewstates.conf.spec
commands.conf.spec              instance.cfg.spec             searchbnf.conf.spec             visualizations.conf.spec
conf_checker.rules              limits.conf.example           segmenters.conf.example         web-features.conf.example
datamodels.conf.example         limits.conf.spec              segmenters.conf.spec            web-features.conf.spec
datamodels.conf.spec            literals.conf.example         server.conf.example             web.conf.example
datatypesbnf.conf.spec          literals.conf.spec            server.conf.spec                web.conf.spec
default-mode.conf.examples      livetail.conf.examples        serverclass.conf.example        wmi.conf.example
default-mode.conf.spec          livetail.conf.spec            serverclass.conf.spec           wmi.conf.spec
default.meta.example            macros.conf.example           serverclass.seed.xml.example    workflow_actions.conf.example
default.meta.spec               macros.conf.spec              serverclass.seed.xml.spec       workflow_actions.conf.spec
deployment.conf.spec            messages.conf.example         setup.xml.spec                  workload_policy.conf.example
deploymentclient.conf.example   messages.conf.spec            source-classifier.conf.example  workload_policy.conf.spec
deploymentclient.conf.spec      metric_alerts.conf.example    source-classifier.conf.spec     workload_pools.conf.example
distsearch.conf.example         metric_alerts.conf.spec       sourcetypes.conf.example        workload_pools.conf.spec
distsearch.conf.spec            metric_rollups.conf.example   sourcetypes.conf.spec           workload_rules.conf.example
event_renderers.conf.example    metric_rollups.conf.spec      splunk-launch.conf.spec         workload_rules.conf.spec
event_renderers.conf.spec       migration.conf.spec           tags.conf.example
eventdiscoverer.conf.example    multikv.conf.example          tags.conf.spec
```

3. View the files in the default directory:

```
ls default
```

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

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/system$ ls default
alert_actions.conf   datamodels.conf       global-banner.conf  metric_rollups.conf   server.conf             ui-tour.conf
app.conf             datatypesbnf.conf     health.conf         multikv.conf          serverclass.conf        viewstates.conf
audit.conf           default-mode.conf     indexes.conf        outputs.conf          source-classifier.conf  visualizations.conf
authentication.conf  distsearch.conf       inputs.conf         procmon-filters.conf  sourcetypes.conf        web-features.conf
authorize.conf       event_renderers.conf  limits.conf         props.conf            telemetry.conf          web.conf
collections.conf     eventdiscoverer.conf  literals.conf       restmap.conf          times.conf              workflow_actions.conf
commands.conf        eventtypes.conf       livetail.conf       savedsearches.conf    transactiontypes.conf   workload_policy.conf
conf.conf            federated.conf        messages.conf       searchbnf.conf        transforms.conf         workload_pools.conf
data                 fields.conf           metric_alerts.conf  segmenters.conf       ui-prefs.conf           workload_rules.conf
```

4. View the files in the local directory:

```
ls local
```

You should see a much shorter list of Splunk configuration (<filename>.conf) files, such as inputs.conf, authentication.conf, server.conf, and web.conf. These are the configuration files that have been modified through initial Splunk installation and configuration.

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/system$ ls local
README  authorize.conf  global-banner.conf  inputs.conf  migration.conf  server.conf  web.conf
```

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

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

```
############################################################################
# General Server Configuration
############################################################################
[general]
serverName = <ASCII string>
* The name that identifies this Splunk software instance for features such as
  distributed search.
* Cannot be an empty string.
* Can contain environment variables.
* After any environment variables are expanded, the server name
  (if not an IPv6 address) can only contain letters, numbers, underscores,
  dots, and dashes. The server name must start with a letter, number, or an
  underscore.
* Default: $HOSTNAME
```

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

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

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

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

```
ls /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps
```

You should see a README file. This directory is currently otherwise empty, however you will use this

location to deploy apps to Splunk deployment clients later in this course.

7. View the README file in the deployment-apps directory.

```
more /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps/README
```

The README file states:

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

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

Notes About Security and Splunk 9.x

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

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

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

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

In a production environment the use of security certificates prevents these warning messages from occurring. Splunk and certificate configuration for validation of Splunk commands is documented under the section "Configure TLS host name validation for the Splunk CLI" at: https://docs.splunk.com/Documentation/Splunk/latest/Security/EnableTLSCertHostnameValidation. This page also discusses enabling certification and hostname validation for other parts of the Splunk environment.

Task 4: Create the outputs base app and enable listening on the indexer

In this exercise, you configure the outputs base app that will be used on universal forwarder #1 and your heavy forwarder to send data to your indexers and validate the receipt of internal splunkd data using your search head.

1. Ensure you are in the deployment-apps directory:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps
```

2. Create a new folder within the deployment-apps directory named all_fowarder_outputs:

```
mkdir all_forwarder_outputs
```

3. Within the all_forwarder_outputs directory, create the metadata and local directories:

```
cd all_forwarder_outputs
mkdir metadata
mkdir local
```

4. Within the metadata directory, create a new file named local.meta with the following configuratio::

```
cd metadata
vi local.meta
```

Press **i** to enter Insert mode.

```
[]
access = read : [ * ], write : [ admin ]
export = system
```

5. Save and close the file. Press Esc to exit Insert mode, then type :wq and press Enter to save and close the file.
6. Within the local directory, create a new file named outputs.conf with the following configuration, replacing <server_one> with the IP address of the indexer you have been assigned:

```
cd ../local
vi outputs.conf
```

Press **i** to enter Insert mode.

```
# BASE SETTINGS

[tcpout]
defaultGroup = primary_indexer

forceTimebasedAutoLB = true
```

```
forwardedindex.0.whitelist = (_audit|_introspection|_internal)

[tcpout:primary_indexer]
server = <server_one>:9997
```

7. Save and close the file.

Note: You can use `hostname -I/ifconfig` to find the host IP address to replace the <server_one> value. You can also reference the CSV file generated for the workshop.

8. Within the local directory, create a new file named limits.conf with the following configuration:

```
vi limits.conf
```

Press **i** to enter Insert mode.

```
# By default a universal or light forwarder is limited to 256kB/s
# Either set a different limit in kB/s, or set the value to zero to
# have no limit.
# Note that a full speed UF can overwhelm a single indexer.

# [thruput]
# maxKBps = 0
```

9. Save and close the file.

Note: We are exposing the limits.conf file as part of this base app to raise awareness of the existence of this file and its purpose. For many customers, the default setting is sufficient, though if this is an intermediate forwarder, the recommendation would be to set this to 0 for unlimited.

10. Within the local directory, create a new file named app.conf with the following configuration:

```
vi app.conf
```

Press **i** to enter Insert mode.

```
[install]
state = enabled

[package]
check_for_updates = false

[ui]
is_visible = false
```

11. Save and close the file.

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/deployment-apps$ ls
README  all_forwarder_outputs
```

12. Navigate to the indexer deployment and create a new inputs app and the inputs.conf

```
cd /opt/showlab/s4a-fm-idx/splunk/etc/apps
mkdir s4a-fm-inputs
cd s4a-fm-inputs
mkdir local
cd local/
echo -e '[splunktcp://9997]\nconnection_host = dns' > inputs.conf
```

13. Restart the indexer instance

```
/opt/showlab/s4a-fm-idx/splunk/bin/splunk restart
```

Task 5: Create the deployment client base app

In this exercise, you configure the deployment client base app that will be used on forwarders to manage configurations remotely.

1. Ensure you are in the deployment-apps directory:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps
```

2. Create a new folder within the deployment-apps directory named all_deploymentclient:

```
mkdir all_deploymentclient
```

3. Within the all_deploymentclient directory, create the metadata and local directories:

```
cd all_deploymentclient
mkdir metadata
mkdir local
```

4. Within the metadata directory, create a new file named local.meta with the following configuration:

```
cd metadata
vi local.meta
```

Press **i** to enter Insert mode.

```
[]
access = read : [ * ], write : [ admin ]
export = system
```

5. Save and close the file.
6. Within the local directory, create a new file named app.conf with the following configuration:

```
cd ../local
vi app.conf
```

Press **i** to enter Insert mode.

```
[install]
state = enabled

[package]
check_for_updates = false

[ui]
is_visible = false
```

6. Save and close the file.
7. Within the local directory, create a new file named deploymentclient.conf with the following configuration (update the targetUri to the IP address of the DS):

```
vi deploymentclient.conf
```

Press **i** to enter Insert mode.

```
[deployment-client]
# Set the phoneHome
# 10 minutes
# phoneHomeIntervalInSecs = 600

[target-broker:deploymentServer]
# Change the targetUri
targetUri = https://<ds-ipaddress>:8091
```

8. Save and close the file.

NOTE: We are using IP address in the workshop, in reality you would want to use a DNS name or CNAME for the deployment server.

HINT: You will need to use ifconfig (locally) or nslookup (not from the local machine because you will get the local IP) to identify the deployment server ip address. You can also use hostname -I or reference the CSV file that as provided for the workshop.

```
splunk@Domane-Demo-i-082c55c8ad0b6226f:/opt/splunk-ds/splunk/etc/deployment-apps$ ls
README  all_deploymentclient  all_forwarder_outputs
```

Task 6: Create the serverclass.conf file

1. Navigate to the serverclass.conf location:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/system/local
```

2. Create/Edit serverclass.conf:

```
vi serverclass.conf
```

3. Define the Serverclass all_forwarders by adding the following configuration to include UF1 and HF1 in the whitelist and assign the app all_forwarder_outputs:

   Press **i** to enter Insert mode.

```
[serverClass:all_forwarders]
restartSplunkWeb = 0
restartSplunkd = 1
stateOnClient = enabled
whitelist.0 = HF1
whitelist.1 = UF1

[serverClass:all_forwarders:app:all_forwarder_outputs]
```

4. Save and close the file.

   NOTE: Use the name defined in the server.conf file of the UF and HF installs

5. Define the Serverclass all_deploymentclient, reopen the serverclass.conf file if it is closed:

```
vi serverclass.conf
```

6. Add the following configuration to include all deployment clients and assign the app all_deploymentclient:

```
[serverClass:all_deploymentclient]
restartSplunkWeb = 0
restartSplunkd = 1
stateOnClient = enabled
whitelist.0 = *

[serverClass:all_deploymentclient:app:all_deploymentclient]
```

7. Press Esc to exit Insert mode, then type :wq and press Enter to save and close the file.
8. Verify the Configuration:
9. To ensure there are no syntax errors in the configuration file, you can use the splunk btool command:

```
/opt/showlab/s4a-fm-ds/splunk/bin/splunk btool serverclass list --debug
```

10. Restart the Splunk Deployment Server:

```
/opt/showlab/s4a-fm-ds/splunk/bin/splunk restart
```

11. Confirm Deployment by logging into the DS Splunk Web UI.
12. Navigate to the "Forwarder Management" section under the "Settings" menu.
13. Verify that the serverclasses all_forwarders and all_deploymentclient are listed.

Task 7: Connect UF1 and HF1 to the Deployment Server for remote management

1. Change directory to where the all_deploymentclient app is located:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps
```

2. On the UF of HF, use hostname -I to find the host's internal IP address for the following commands.
3. Use the scp command to copy the all_deploymentclient app to UF1's apps directory:

```
scp -r -P 2222 all_deploymentclient/
splunk@{IPAddress}:/opt/showlab/s4a-fm-uf/splunkforwarder/etc/apps
```

Replace IPAddress with the actual username for SSH access to UF1.

You will be prompted to enter the password for the SSH user on UF1.

4. Use the scp command to copy the all_deploymentclient app to HF1's apps directory:

```
scp -r -P 2222 all_deploymentclient/
splunk@{IPAddress}:/opt/showlab/s4a-fm-hf/splunk/etc/apps
```

Replace IPAddress with the actual username for SSH access to UF1.

You will be prompted to enter the password for the SSH user on HF1.

5. Verify the Copy on UF1 by SSHing into it

```
ssh -p 2222 splunk@<ip_address>
```

6. Navigate to the apps directory and list the contents to verify the app is copied:

```
cd /opt/showlab/s4a-fm-uf/splunkforwarder/etc/apps/
ls -l
```

7. Navigate to the server.conf to update the clientname field

```
vi ../system/local/server.conf
```

8. Update the clientname to UF1

```
[general]
serverName = UF1
```

9. Remove legacy indexers

```
rm -rf /opt/showlab/s4a-fm-uf/splunkforwarder/etc/system/local/outputs.conf
```

10. Restart splunkd so the agent will phone home to the deployment server.

Note: If you are getting an error indicating ‘remote login has been disabled’, get around this by setting allowRemoteLogin = always in system/local/server.conf

```
/opt/showlab/s4a-fm-uf/splunkforwarder/bin/splunk restart
```

11. Verify the Copy on the HF instance by navigating to it.

```
cd /opt/showlab/s4a-fm-hf/splunk/etc/apps/
ls -l
```

12. Navigate to the server.conf to update the clientname field

```
vi ../system/local/server.conf
```

13. Update the clientname to HF1

```
[general]
serverName = HF1
```

14. Restart splunkd so the agent will phone home to the deployment server.

```
/opt/showlab/s4a-fm-hf/splunk/bin/splunk restart
```

15. Confirm Deployment by logging into the DS Splunk Web UI.
16. Navigate to the "Forwarder Management" section under the "Settings" menu.
17. Verify that the serverclasses all_forwarders and all_deploymentclient are listed and that the corresponding apps are deployed to the both the UF1 and HF1 hosts.
18. Navigate to the Search Head UI and validate you are receiving the _internal data for both UF1 and HF1

Task 8: Create an inputs app and deploy it from the Deployment server.

1. Open a terminal and SSH into the Splunk deployment server:

```
ssh username@deployment-server
```

2. Change the directory to the deployment-apps directory where Splunk apps are managed:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/deployment-apps
```

3. Create a new directory for the app named all_linux_inputs:

```
mkdir all_linux_inputs
```

4. Within the all_linux_inputs directory, create the metadata and local directories:

```
cd all_linux_inputs
mkdir metadata
mkdir local
```

5. Within the metadata directory, create a new file named local.meta with the following configuration:

```
cd metadata
vi local.meta
```

Press **i** to enter Insert mode.

```
[]
access = read : [ * ], write : [ admin ]
export = system
```

6. Save and close the file.
7. Within the local directory, create a new file named app.conf with the following configuration:

```
cd ../local
vi app.conf
```

Press **i** to enter Insert mode.

```
[install]
state = enabled

[package]
check_for_updates = false

[ui]
is_visible = false
```

8. Save and close the file.
9. Within the local directory, create a new file named inputs.conf with the following configuration:

```
vi inputs.conf
```

Press **i** to enter Insert mode.

```
[journald://default]
journalctl-include-fields = PRIORITY,CMD,EXE
disabled = false
index = linux
sourcetype = journald
```

10. Save and close the file.
11. Change directory to the system/local directory to edit serverclass.conf:

```
cd /opt/showlab/s4a-fm-ds/splunk/etc/system/local
```

12. Open the serverclass.conf file in vim:

```
vi serverclass.conf
```

13. Add the following configuration to create the all_linux serverclass and whitelist Linux operating systems using the machinetypes filter:

```
[serverClass:all_linux]
restartSplunkWeb = 0
```

```
restartSplunkd = 1
stateOnClient = enabled
whitelist.0 = *
machineTypesFilter = linux-i686, linux-x86_64

[serverClass:all_linux:app:all_linux_inputs]
```

14. Save and close the file.
15. To ensure there are no syntax errors in the configuration file, use the splunk btool command:

```
/opt/showlab/s4a-fm-ds/splunk/bin/splunk btool serverclass list --debug
```

16. For the changes to take effect, restart the Splunk deployment server:

```
/opt/showlab/s4a-fm-ds/splunk/bin/splunk restart
```

17. Log in to the Splunk Web interface.
18. Navigate to the "Forwarder Management" section under the "Settings" menu.
19. Verify that the all_linux serverclass is listed and that the all_linux_inputs app is deployed to the specified clients.

Extra Credit Task 1: Create a universal forwarder tarball image.

Note: **<u>There is not another instance to copy the tarball to for this lab. This is for demonstration purposes only.</u>**

20. Open a terminal and SSH into UF1:

```
ssh splunk@UF1
```

Replace username with your actual SSH username and UF1 with the hostname or IP address of UF1.

21. Change directory to the splunkforwarder bin directory:

```
cd /opt/showlab/s4a-fm-uf/splunkforwarder/bin
```

22. Ensure boot-start is enabled

```
sudo ./splunk enable boot-start
```

23. Stop the Splunkforwarder Service:

```
sudo ./splunk stop
```

24. Run the clone-prep-clear-config script to prepare the instance for cloning:

```
sudo ./splunk clone-prep-clear-config
```

25. Change directory to the parent directory of splunkforwarder:

```
cd /opt/showlab/s4a-fm-uf/
```

26. Use the tar command to create a compressed tarball file of the splunkforwarder directory:

```
sudo tar -czvf splunkforwarder_image.tar splunkforwarder
```

This command will create a tarball named splunkforwarder_image.tar.gz in the /opt/showlab/s4a-fm-uf directory.

27. SCP the image to UF2

```
scp -P 2222 /opt/showlab/s4a-fm-uf/splunkforwarder_image.tar.gz username@UF2:/tmp
```

28. Restart the Splunkfowarder service

```
/opt/showlab/s4a-fm-uf/splunkforwarder/bin/splunk start
```

29. Verify the UF1 instance is running and continuing to send data to the indexers.

30. SSH to UF2 instance

```
ssh splunk@UF2
```

Replace username with your actual SSH username and UF2 with the hostname or IP address of UF2.

31. Navigate to the /tmp Directory:

```
cd /tmp
```

32. Verify the Tarball File

```
ls -l
```

Ensure the splunkforwarder_image.tar.gz file is listed.

33. Use the tar command to extract the contents of the tarball into the /opt directory:

```
tar -xzvf splunkforwarder_image.tar.gz -C /opt/showlab/s4a-fm-uf
```

34. List the contents of the /opt directory to verify the extraction:

```
ls -l /opt/showlab/s4a-fm-uf
```

You should see the splunkforwarder directory and its contents extracted into /opt.

35. Start the Splunkfowarder service

```
sudo /opt/showlab/s4a-fm-uf/splunkforwarder/bin/splunk start
```

36. Confirm Deployment by logging into the DS Splunk Web UI.
37. Navigate to the "Forwarder Management" section under the "Settings" menu.
38. Verify that the serverclasses all_forwarders and all_deploymentclient are listed and that the corresponding apps are deployed to all UF and HF hosts.
39. Navigate to the Search Head UI and validate you are receiving the _internal data for UF1, UF2, and HF1

Useful Deployment Server Commands

From the Deployment Server (DS):

splunk reload deploy-server	Checks all apps for changes and notifies the relevant clients the next time they phone home
splunk list deploy-clients	Displays information about the deployment clients

From the Deployment Client:

splunk set deploy-poll	Connects the client to the deployment server and management port
splunk show deploy-poll	Displays the current deployment server and management port
splunk list forward-server	Displays the current forward server configuration
splunk disable deploy-client	Disables the deployment client

Links

About deployment server and forwarder Management	https://docs.splunk.com/Documentation/Splunk/latest/Updating/Aboutdeploymentserver
Plan a deployment	https://docs.splunk.com/Documentation/Splunk/latest/Updating/Planadeployment
Deploy apps to clients	https://docs.splunk.com/Documentation/Splunk/latest/Updating/Updateconfigurations
Using serverclass.conf	https://docs.splunk.com/Documentation/Splunk/latest/Updating/Useserverclass.conf
serverclass.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/Serverclassconf
outputs.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/outputsconf
inputs.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/inputsconf
app.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/appconf
local.meta	https://docs.splunk.com/Documentation/Splunk/latest/Admin/Defaultmetaconf
props.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/propsconf
transforms.conf	https://docs.splunk.com/Documentation/Splunk/latest/Admin/transformsconf
General Splunk Naming Conventions	Naming conventions - Splunk Lantern
Knowledge Object Naming Conventions	Develop naming conventions for knowledge objects - Splunk Documentation
Thoughts on Naming Conventions in Splunk	https://community.splunk.com/t5/Splunk-SOAR-f-k-a-Phantom/Recommendations-for-naming-conventions-and-organization-of/m-p/501574