

Chapter 5 - Universal forwarders

IN THIS CHAPER

Explaining universal forwarders

Installing forwarders on Windows and Linux

Collecting remote logs using forwarders

What are universal forwarders?

The most efficient way to gather data from any remote machine is to install **universal forwarders** on remote hosts. A universal forwarder is a dedicated, lightweight version of Splunk that contains only the essential components needed to send data. It is similar to the Splunk server and it has many similar features, but it does not contain Splunk web and doesn't come bundled with the Python executable and libraries.

Forwarders are configured to consume data and forward it on to Splunk indexers for processing. They can handle exactly the same types of data and can consume the data in the same way as any Splunk instance, with one difference: they do not index the data themselves. Instead, they input the data and refer it to a Splunk indexer, which then does the indexing and searching.

In a typical Splunk deployment, forwarders serve as the primary consumers of data. For example, if you have a number of web servers generating data that you want to be able to search centrally, you can install a Splunk indexer and then install forwarders on all web servers. The forwarders can then be configured to send the logs to the indexer, which will store them and make them available for searching.

NOTE

Besides universal forwarders, two other types of forwarders exist in Splunk:

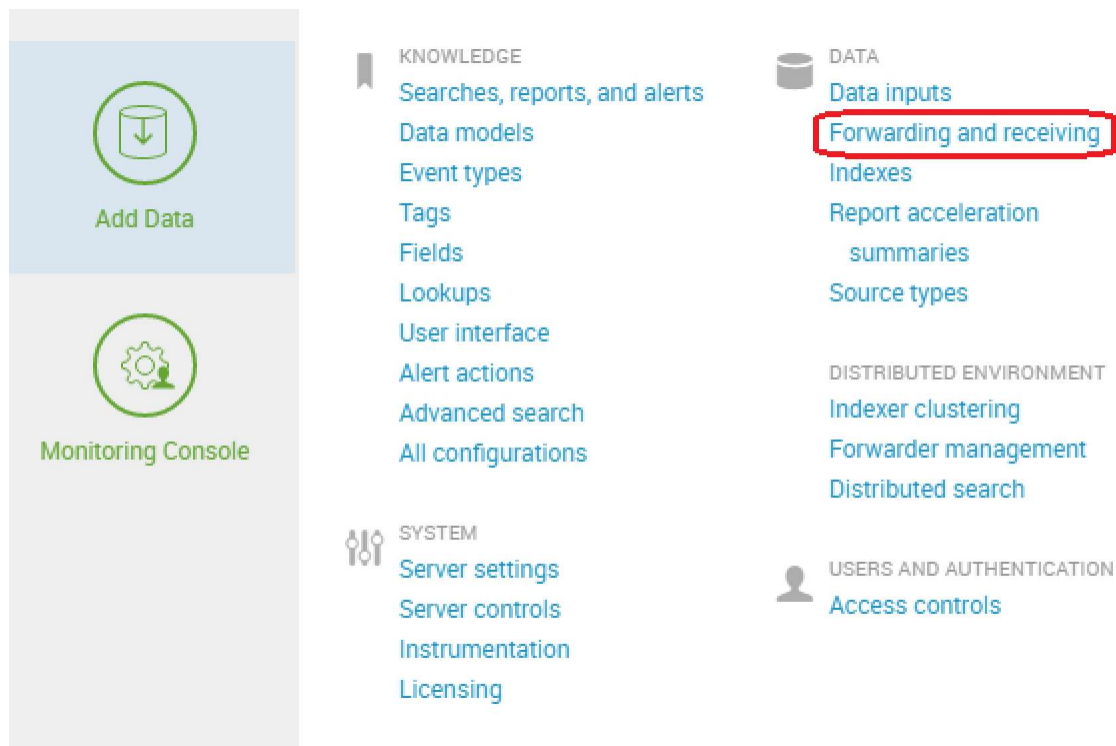
heavy forwarder - a full Splunk Enterprise instance that can index, search, and change data as well as forward it. Heavy forwarders have most of the capabilities that indexers have, except that they cannot perform distributed searches.

light forwarder - a full Splunk Enterprise instance, with most of the features disabled to achieve a small resource footprint. The light forwarders have been deprecated as of Splunk Enterprise version 6.0 and are superseded by universal forwarders.

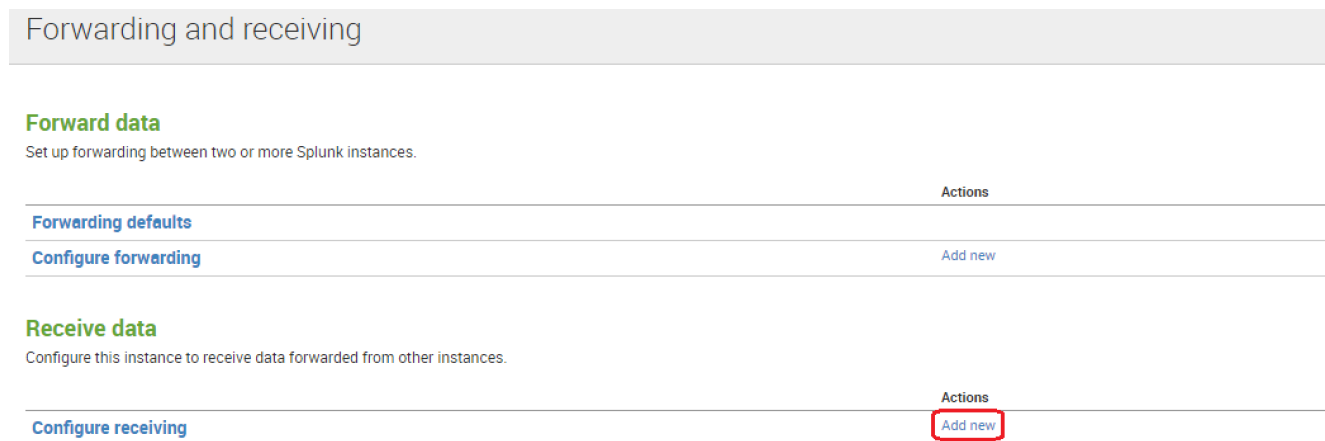
Set up a receiver

To collect logs from remote machines, you need to configure both a receiver and a forwarder. The **receiver** is the Splunk instance that will receive the data sent by the forwarder. The receiver is usually a Splunk indexer or another forwarder configured to receive data from forwarders.

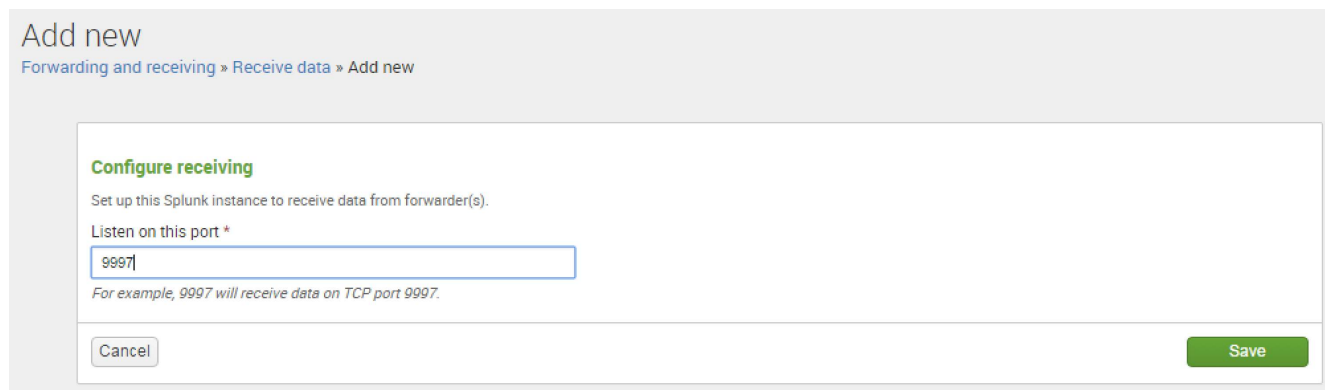
You can use **Splunk Web** to set up a Splunk instance to serve as a receiver. Log in to Splunk Web using the administrative account and go to **Settings > Forwarding and Receiving**:



Click **Add new** under the **Receive data** section:



Specify the TCP port that you want the receiver to listen on. The port is usually **9997**:



And that's it! The receiver has been enabled and we can now configure forwarders to send data to it.

NOTE

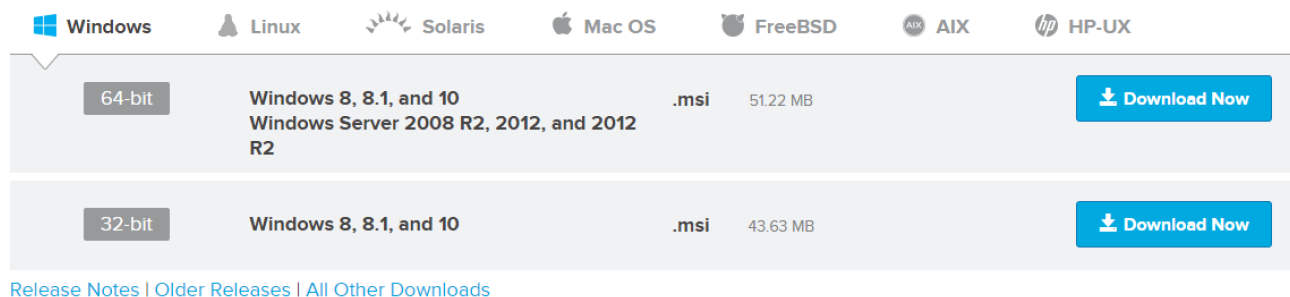
Depending on the Splunk version, you might need to restart Splunk to apply the changes.

Install a Splunk forwarder on Windows

As you've already learned, a universal forwarder is a dedicated, lightweight version of Splunk that sends logs from a remote host to the indexer. To install a universal forwarder, you need to download it first. Go to https://www.splunk.com/en_us/download/universal-forwarder.html and choose the forwarder for your operating system:

Splunk Universal Forwarder 6.5.2

Universal Forwarders provide reliable, secure data collection from remote sources and forward that data into Splunk software for indexing and consolidation. They can scale to tens of thousands of remote systems, collecting terabytes of data.

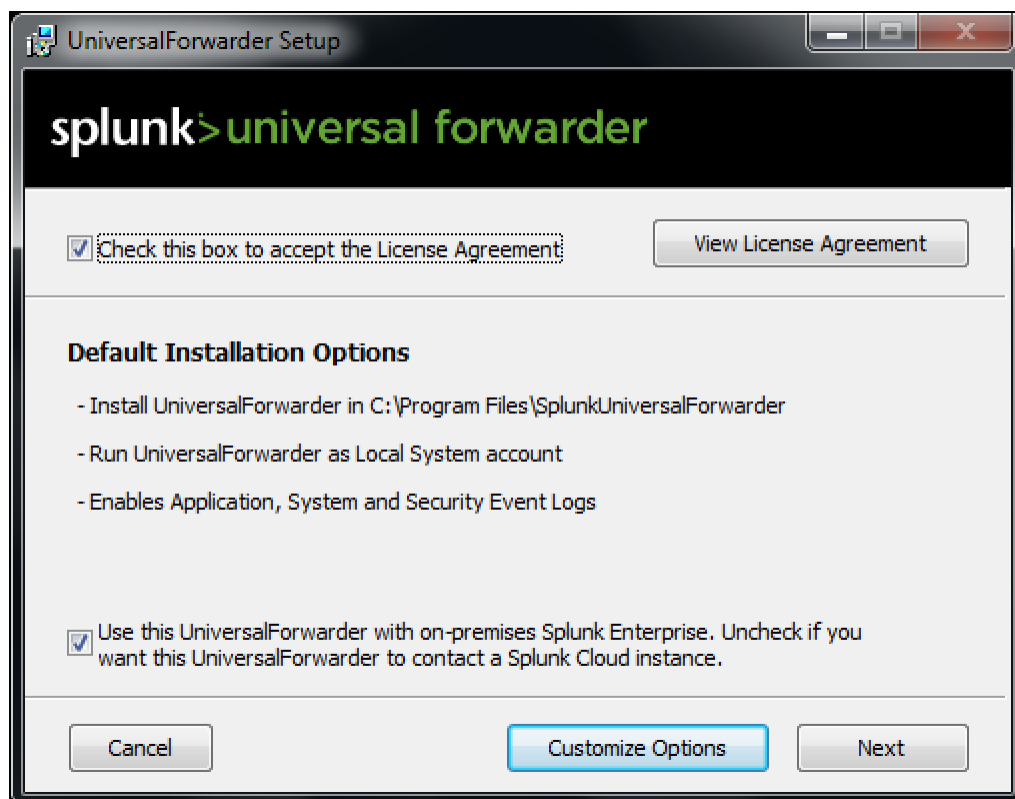


Operating System	Architecture	Version	File Format	Size	Action
Windows	64-bit	Windows 8, 8.1, and 10 Windows Server 2008 R2, 2012, and 2012 R2	.msi	51.22 MB	Download Now
Windows	32-bit	Windows 8, 8.1, and 10	.msi	43.63 MB	Download Now

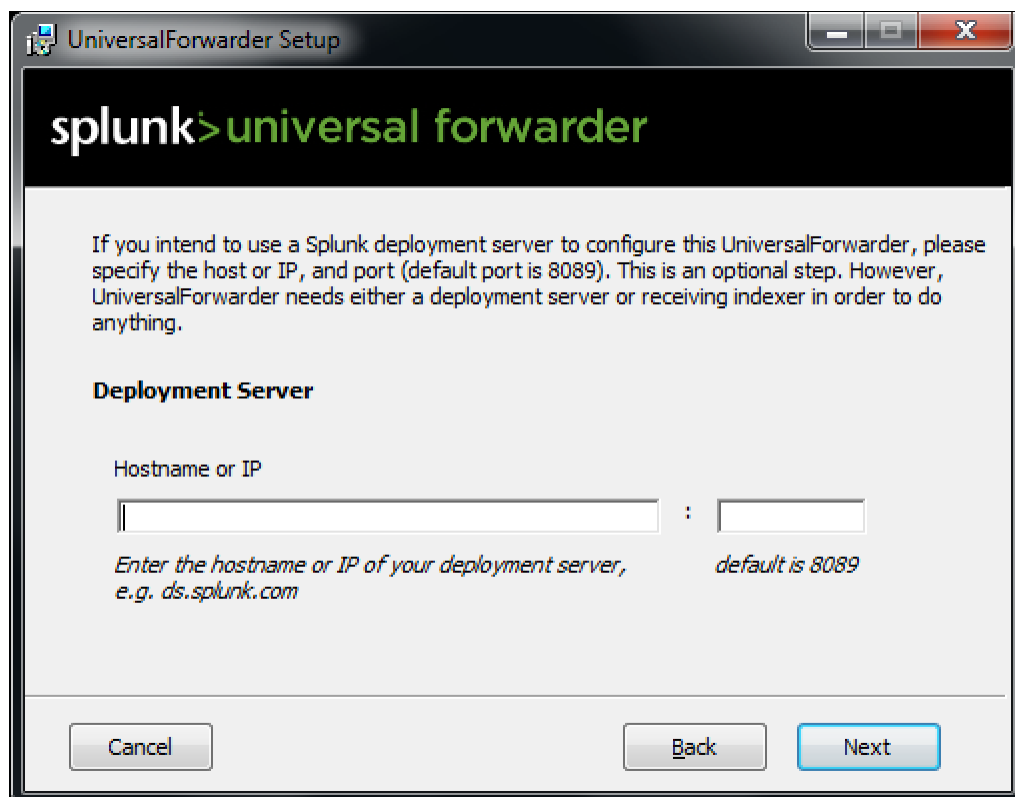
[Release Notes](#) | [Older Releases](#) | [All Other Downloads](#)

I will choose the Windows 64-bit version of the forwarder and show you how you can install it on Windows 7.

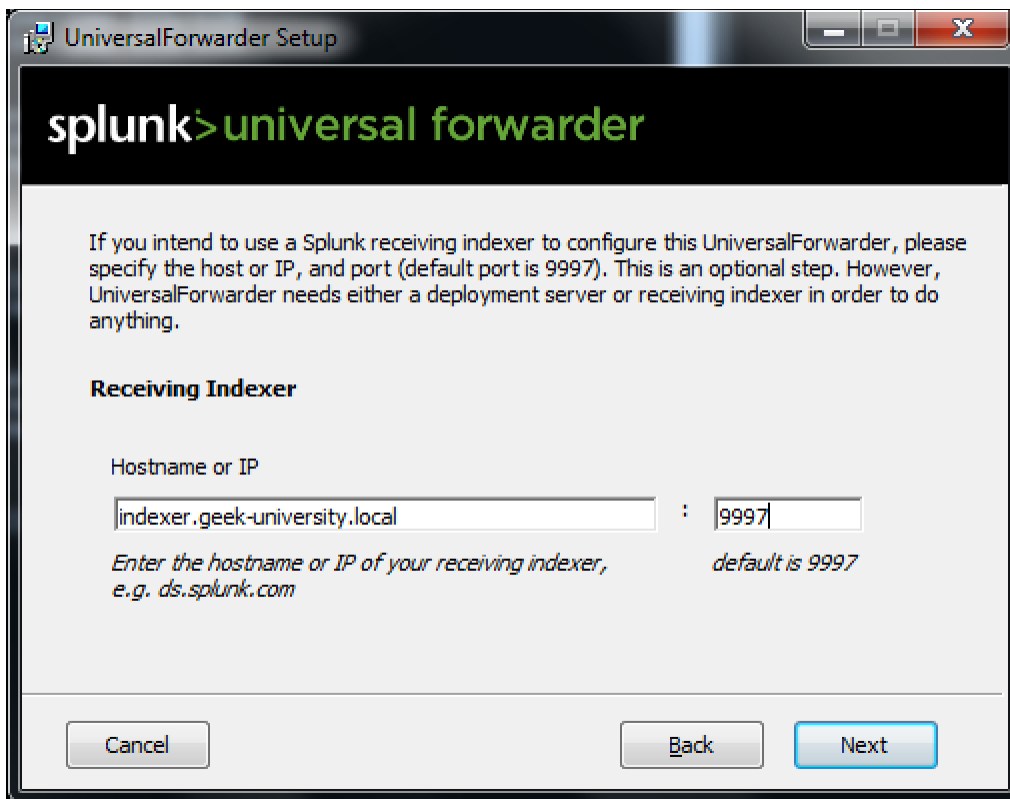
You can start the installation by double-clicking the installer file. You should be greeted with the **Setup page**, on which you can accept the default options or customize the options. By default, the universal forwarder will be installed in **C:\Program Files\SplunkUniversalForwarder**, use a local system account, and collect the **Application**, **System**, and **Security** Windows Event logs:



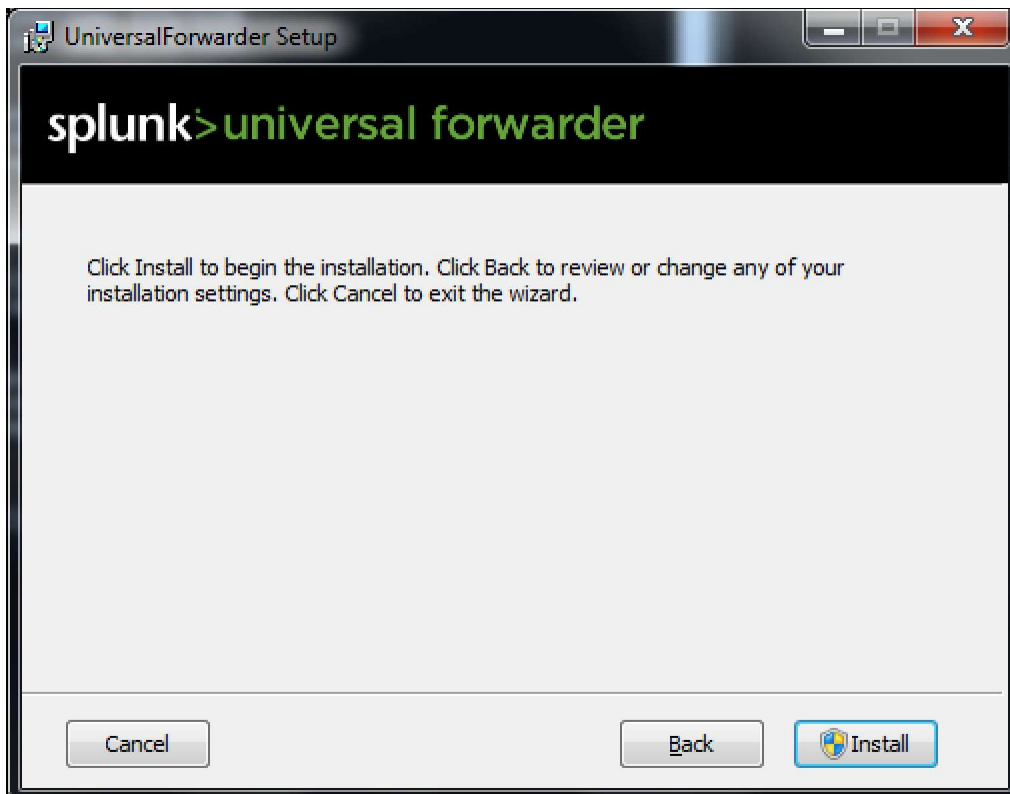
Next, you need to enter the hostname or IP address and management port of your deployment server (the default management port is **8089**). The deployment server can be used to push configuration updates to the universal forwarder. Note that this is an optional step; if you skip it, you should enter the receiving indexer in the next step.



Next, enter the hostname or IP address and the receiving port of your **indexer** (the default port is **9997**):



Click **Install** to start the installation:



Once the installation is complete, the universal forwarder should automatically start.

Starting and stopping universal forwarders

The forwarder needs to be started in order to forward data to the indexer. You also need to restart the forwarder each time you've made changes to the configuration files. On Windows, you can control the forwarder's state using the Services MMC. It is also possible to start and stop forwarders using the command line on either a Windows or a Linux host.

To start a forwarder, open Command Prompt, browse to the **\$SPLUNK_HOME\bin** folder, and execute the `.\splunk start` command:

```
C:\Program Files\SplunkUniversalForwarder\bin>.\splunk start
```

To stop a forwarder, simply use the `stop` keyword:

```
C:\Program Files\SplunkUniversalForwarder\bin>.\splunk stop
```

To restart a forwarder, use the `restart` keyword (do this after each change in the configuration files):

```
C:\Program Files\SplunkUniversalForwarder\bin>.\splunk restart
```

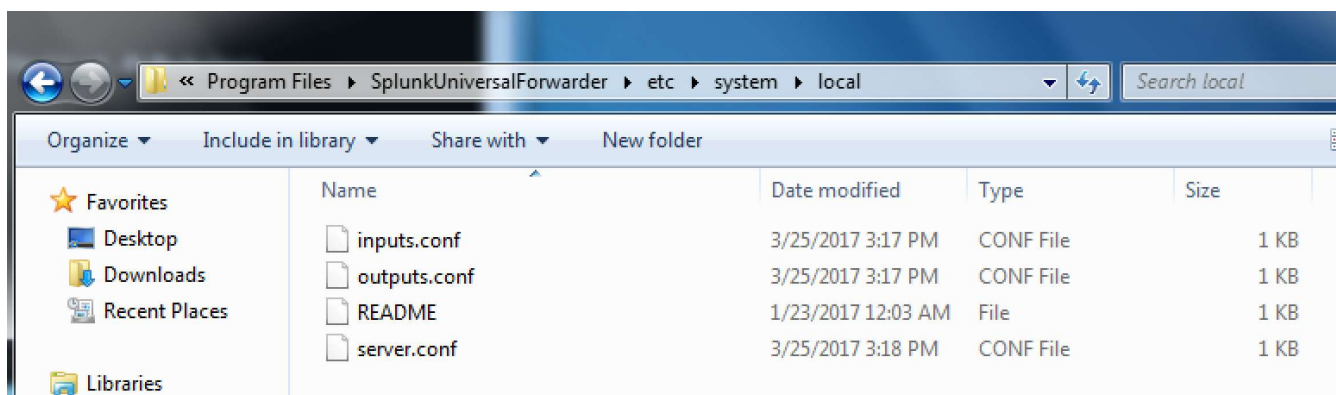
NOTE

On a Linux host, simply use `./splunk` instead of `.\splunk` when executing the commands above.

Monitor logs using forwarders

To define which logs will be monitored and forwarded to the indexer, you need to edit the **inputs.conf** file in the **\$SPLUNK_HOME\etc\system\local** directory. Here is how it can be done on Windows:

First, browse to the directory on the forwarder containing the **inputs.conf** file. In my case, this will be the **C:\Program Files\SplunkUniversalForwarder\etc\system\local** directory:



Open the **inputs.conf** file in a text editor:

```
[default]
host = splunk-forwarder
```

Now, we need to add the data inputs by specifying the **stanzas**. A stanza is a section of a configuration file that begins with a text string enclosed in brackets and contains one or more configuration parameters defined by the key/value pairs. I want to monitor the log file located at **C:\logs\secure.log**, classify its events as the sourcetype of **secure_access_logs**, and store them into the index called **secure_logs**:

[default]

host = splunk-forwarder

[monitor://C:\logs\secure.log]

sourcetype = secure_access_logs

index = secure_logs

NOTE

If you can't save the changes, re-open the file as administrator.

As you can see from the example above, I've specified the file to monitor using the *[monitor://path]* syntax. I've also specified the sourcetype and index for the log.

After we've added the inputs, we need to restart the forwarder in order to apply the changes. I can search the logs on the indexer to make sure that the events have been received and indexed:

The screenshot shows the Splunk Search & Reporting interface. The search bar contains 'index=secure_logs' and the results show 9,983 events. The interface includes tabs for Search, Datasets, Reports, Alerts, and Dashboards. The search results are displayed in a table format with columns for Time and Event. The table shows several failed password attempts for various users (administrator, appserver, gitosis, djohnson, email, root) from different IP addresses (207.36.232.245, 59.5.100.202, 10.3.10.46) on March 13, 2016.

i	Time	Event
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[1370]: Failed password for invalid user administrator from 207.36.232.245 port 4438 ssh2 host = splunk-forwarder source = C:\logs\secure.log sourcetype = secure_access_logs
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[5352]: Failed password for invalid user appserver from 59.5.100.202 port 4512 ssh2 host = splunk-forwarder source = C:\logs\secure.log sourcetype = secure_access_logs
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[5951]: Failed password for invalid user gitosis from 59.5.100.202 port 1277 ssh2 host = splunk-forwarder source = C:\logs\secure.log sourcetype = secure_access_logs
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[57740]: Accepted password for djohnson from 10.3.10.46 port 9507 ssh2 host = splunk-forwarder source = C:\logs\secure.log sourcetype = secure_access_logs
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[2004]: Failed password for invalid user email from 59.5.100.202 port 1296 ssh2 host = splunk-forwarder source = C:\logs\secure.log sourcetype = secure_access_logs
>	3/13/16 12:15:03.000 AM	Thu Mar 13 2016 00:15:03 www3 sshd[1638]: Failed password for root from 59.5.100.202 port 2281 ssh2

Monitor remote Windows event logs

Since I've installed a forwarder on a Windows machine, I can edit the **inputs.conf** file to configure Windows event logs I want to monitor. Here is the configuration that will allow us to monitor the **Windows Security**, **Application**, and **System** event logs and store them in the index called *remote_windows_logs*:

```
[WinEventLog://Application]
index=remote_windows_logs
```

```
[WinEventLog://Security]
index=remote_windows_logs
```

```
[WinEventLog://System]
index=remote_windows_logs
```

Restart the forwarder in order for the changes to take effect. We can run a search on our Splunk indexer to verify that events have indeed been indexed:

The screenshot shows the Splunk Search & Reporting interface. At the top, there's a green navigation bar with tabs for Search, Datasets, Reports, Alerts, and Dashboards. The 'Search' tab is active. Below the navigation bar, there's a search bar with the text 'index=remote_windows_logs'. To the right of the search bar are buttons for 'Save As' and 'Close'. Below the search bar, there's a status bar showing '5,008 events (before 3/28/17 6:59:59.000 PM)' and 'No Event Sampling'. Below the status bar, there's a timeline visualization showing a single bar for the date 3/28/17. Below the timeline, there's a table of event results. The table has columns for 'Time' and 'Event'. The first row shows an event from 3/28/17 6:59:42.000 PM with LogName=System, SourceName=Microsoft-Windows-Service Control Manager, and EventCode=7036. The second row shows an event from 3/28/17 6:59:33.000 PM with LogName=System, SourceName=Microsoft-Windows-Service Control Manager, and EventCode=7036. The third row shows an event from 3/28/17 6:44:52.000 PM with LogName=System, SourceName=Microsoft-Windows-Service Control Manager, and EventCode=7036. The table also includes a 'Selected Fields' section on the left with fields like host, source, and sourcetype. The 'Interesting Fields' section on the left includes fields like ComputerName, EventCode, EventType, index, Keywords, linecount, LogName, Message, OpCode, punct, and RecordNumber.

Time	Event
3/28/17 6:59:42.000 PM	03/28/2017 06:59:42 PM LogName=System SourceName=Microsoft-Windows-Service Control Manager EventCode=7036 EventType=4 host = splunk-forwarder source = WinEventLog:System sourcetype = WinEventLog:System
3/28/17 6:59:33.000 PM	03/28/2017 06:59:33 PM LogName=System SourceName=Microsoft-Windows-Service Control Manager EventCode=7036 EventType=4 host = splunk-forwarder source = WinEventLog:System sourcetype = WinEventLog:System
3/28/17 6:44:52.000 PM	03/28/2017 06:44:52 PM LogName=System SourceName=Microsoft-Windows-Service Control Manager EventCode=7036 EventType=4 host = splunk-forwarder source = WinEventLog:System sourcetype = WinEventLog:System

Notice how Splunk automatically extracted the source and sourcetype fields:

source

3 Values, 100% of events

Selected Yes No

Reports

Top values Top values by time Rare values

Events with this field

Values	Count	%
WinEventLog:System	3,070	61.302%
WinEventLog:Security	989	19.748%
WinEventLog:Application	949	18.95%

host = splunk-forwarder | source = WinEventLog:System | sourcetype = WinEventLog:Syste

> 3/28/17 03/28/2017 06:44:52 PM LogName=System SourceName=Microsoft-Windows-Service Control Manager EventCode=7036 EventType=4

Show all 12 lines

Install a Splunk forwarder on Linux

You can install a Splunk forwarder on your Linux using one of the following methods:

- using a Splunk forwarder .tar file
- using a Splunk forwarder .deb file
- using a Splunk forwarder .rpm file

In this section we will show you how to install a Splunk forwarder on Ubuntu, which is a popular Debian-based Linux distribution. The first thing we need to do is to download the software. Go to https://www.splunk.com/en_us/download/universal-forwarder.html and click the **Linux** button. Since Ubuntu is .deb based, I will choose the .deb version of the software to download:

Splunk Universal Forwarder 6.5.2

Universal Forwarders provide reliable, secure data collection from remote sources and forward that data into Splunk software for indexing and consolidation. They can scale to tens of thousands of remote systems, collecting terabytes of data.

Windows	Linux	Solaris	Mac OS	FreeBSD	AIX	HP-UX									
<div>64-bit</div> <div>2.6+ kernel Linux distributions</div> <table border="1"> <tr> <td>.rpm</td> <td>18.81 MB</td> <td>Download Now</td> </tr> <tr> <td>.deb</td> <td>18.78 MB</td> <td>Download Now</td> </tr> <tr> <td>.tgz</td> <td>18.89 MB</td> <td>Download Now</td> </tr> </table>							.rpm	18.81 MB	Download Now	.deb	18.78 MB	Download Now	.tgz	18.89 MB	Download Now
.rpm	18.81 MB	Download Now													
.deb	18.78 MB	Download Now													
.tgz	18.89 MB	Download Now													

Open the shell and browse to the packet location. Note that .deb version can only be installed in the default location (**/opt/splunk**). To start the installation, run the `sudo dpkg -i splunk_package_name.deb` command:

```
bob@ubuntu:/tmp$ sudo dpkg -i splunkforwarder-6.5.2-67571ef4b87d-linux-2.6-amd64.deb
Selecting previously unselected package splunkforwarder.
(Reading database ... 211624 files and directories currently installed.)
Preparing to unpack splunkforwarder-6.5.2-67571ef4b87d-linux-2.6-amd64.deb ...
Unpacking splunkforwarder (6.5.2) ...
Setting up splunkforwarder (6.5.2) ...
complete
```

To start a Splunk universal forwarder, browse to the **/bin** directory in the **/opt/splunkforwarder/** directory and run the `sudo ./splunk start` command:

```
bob@ubuntu:~$ cd /opt/splunkforwarder/bin
bob@ubuntu:/opt/splunkforwarder/bin$ sudo ./splunk start
```

The first time you start Splunk after a new installation, you will need to accept the license agreement. Press **y** to accept the license and start the forwarder. You can run the `sudo ./splunk status` command to verify that the forwarder is indeed running:

```
bob@ubuntu:/opt/splunkforwarder/bin$ sudo ./splunk status
splunkd is running (PID: 3712).
splunk helpers are running (PIDs: 3713).
```

Configure a Splunk forwarder on Linux

Once we've installed forwarder on Linux, we can configure it to send logs to the indexer. Here are the steps:

From the **/opt/splunkforwarder/bin** directory, run the `sudo ./splunk enable boot-start` command to enable the Splunk auto-start:

```
bob@ubuntu:/opt/splunkforwarder/bin$ sudo ./splunk enable boot-start
[sudo] password for bob:
Init script installed at /etc/init.d/splunk.
Init script is configured to run at boot.
```

Next, you need to configure the indexer that the forwarder will send its data to. This can be done using the shell commands. The `./splunk add forward-server HOST:9997 -auth USERNAME:PASSWORD` command will specify the forwarder the logs will be sent to. The *admin* and *changeme* are the default values for the username and password:

```
bob@ubuntu:/opt/splunkforwarder/bin$ sudo ./splunk add forward-server indexer.geek-
university.local:9997 -auth admin:changeme
Added forwarding to: indexer.geek-university.local:9997.
```

To add the data you would like to consume and send to the indexer, run the `sudo ./splunk add monitor LOG -sourcetype SOURCE_TYPE -index NAME`. For example, to add the **/var/log/syslog** file with the sourcetype of *linux* and store it to the index called *linux_logs*, we would use the following command:

```
bob@ubuntu:/opt/splunkforwarder/bin$ sudo ./splunk add monitor /var/log/syslog -sourcetype linux -index
linux_logs
Added monitor of '/var/log/syslog'.
```

Restart the forwarder to apply the changes (`sudo ./splunk restart`). We can run a search to verify that events are indeed being received by the indexer:

Save As  Close

All time ▾

All time ▾ Q

✓ 212 events (before 3/28/17 9:24:04.000 PM) No Event Sampling ▾

Job Smart Mode

Events (212) Patterns Statistics Visualization

Format Timeline ▾ - Zoom Out + Zoom to Selection × Deselect

1 minute per column

List Format 20 Per Page

< Prev 1 2 3 4 5 6 7 8 9 ... Next >

[← Hide Fields](#)

All Fields

Selected Fields

a host 1

a source 1

```
a sourcetype 1
```

Interesting Fields

```
# date_hour 2
```

```
# date_mday
```

```
# date_minute 10
```

```
a date_month 1
```

```
# date_second 32
```

```
q date_wday 1
```

```
# date_year 1
```

```
a date_zone 1
```

a index 1

```
# linecount 1
```

a punct 99

i	Time	Event
>	3/28/17 9:17:33.000 PM	Mar 28 12:17:33 ubuntu systemd[1]: Started CUPS Scheduler. host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:17:01.000 PM	Mar 28 12:17:01 ubuntu CRON[23665]: (root) CMD (cd / && run-parts --report /etc/cron.hourly) host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:14:02.000 PM	Mar 28 12:14:02 ubuntu systemd[1]: Started ACPI event daemon. host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:14:02.000 PM	Mar 28 12:14:02 ubuntu systemd[1]: Started CUPS Scheduler. host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:14:02.000 PM	Mar 28 12:14:02 ubuntu systemd[1]: apt-daily.timer: Adding 3h 54min 14.518200s random time. host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:14:02.000 PM	Mar 28 12:14:02 ubuntu systemd[1]: Reloading. host = ubuntu source = /var/log/syslog sourcetype = linux
>	3/28/17 9:11:40.000 PM	Mar 28 12:11:40 ubuntu systemd[1]: apt-daily.timer: Adding 3h 55min 41.682051s random time. host = ubuntu source = /var/log/syslog sourcetype = linux

