

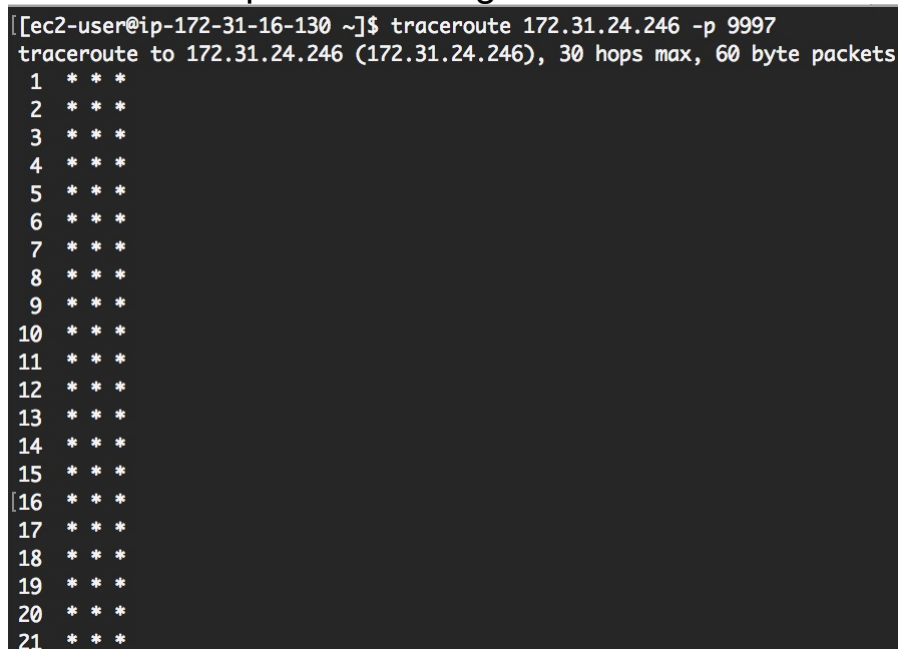
Communication Issues between the Splunk universal forwarder and the Splunk server

1. As a first step, we will check and see if Splunk can use a traceroute to communicate between instances.

1.1. To do this, we will open up terminal and enter the following

```
traceroute <TARGET_IP_ADDRESS> -p 9997
```

1.2. If your firewall is blocking the connections you should see a series of asterisks. The reason for these are due to a timeout in the connection attempt that is being made.



```
[ec2-user@ip-172-31-16-130 ~]$ traceroute 172.31.24.246 -p 9997
traceroute to 172.31.24.246 (172.31.24.246), 30 hops max, 60 byte packets
 1 * * *
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 * * *
 7 * * *
 8 * * *
 9 * * *
10 * * *
11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
```

1.3. A successful attempt would show connections being made to various IP addresses as seen below.

```

[ec2-user@ip-172-31-16-130 ~]$ traceroute 8.8.8.8 -p 9997
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1 * * *
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 100.65.9.161 (100.65.9.161) 0.483 ms 100.65.8.129 (100.65.8.129) 0.303 ms 100.65.10.97 (100.65.10.97) 0.300 ms
 7 52.93.13.64 (52.93.13.64) 0.529 ms 52.93.15.208 (52.93.15.208) 0.501 ms 54.239.48.180 (54.239.48.180) 0.519 ms
 8 52.93.12.98 (52.93.12.98) 2.445 ms 52.93.13.16 (52.93.13.16) 2.661 ms 52.93.12.236 (52.93.12.236) 2.239 ms
 9 52.93.13.49 (52.93.13.49) 0.725 ms 52.93.12.85 (52.93.12.85) 1.926 ms 52.93.13.13 (52.93.13.13) 0.647 ms
10 54.239.43.135 (54.239.43.135) 8.102 ms 54.239.43.137 (54.239.43.137) 8.059 ms 54.239.43.119 (54.239.43.119) 7.608 ms
11 54.239.42.234 (54.239.42.234) 7.134 ms 54.239.42.218 (54.239.42.218) 8.204 ms 52.95.53.134 (52.95.53.134) 12.230 ms
12 52.95.52.228 (52.95.52.228) 10.137 ms 52.95.53.4 (52.95.53.4) 11.776 ms 52.95.52.94 (52.95.52.94) 43.815 ms
13 52.95.52.162 (52.95.52.162) 35.501 ms 52.95.52.123 (52.95.52.123) 8.131 ms 52.95.52.251 (52.95.52.251) 8.004 ms

```

1.4. To fix this, we first need to go into our iptables and enable receiving data from our port.

1.4.1. `sudo iptables -I INPUT -p tcp -s <ip_address> --dport 9997 -j ACCEPT`

1.4.2. `sudo iptables -A OUTPUT -p tcp --sport 9997 -m conntrack --ctstate ESTABLISHED -j ACCEPT`

Universal Forwarder: Receiver Indexer misconfigured in outputs.conf

The `outputs.conf` file defines how forwarders send data to receivers. It is a critical file for configuring forwarders as it addresses where the forwarder should send data to. Configuration settings on the forwarder running on Linux require that you edit `outputs.conf`. You may need to verify the correct address for the receiver/indexer.

1. On your forwarder, open

```
$SPLUNK_HOME/etc/system/local/outputs.conf.
```

2. Verify the address and port are correct. This is an example syntax if you're defining a single-server stanza (single indexer):

```
[tcpoutserver://<ipaddress_or_servername>:<port>]
disabled = false
```

The `ipaddress_or_servername` is the address of your Splunk indexer, and the `port` is the receiving port on the Splunk indexer (usually 9997).

3. If you make and save any changes, restart the Splunk service on the forwarder.

`$SPLUNK_HOME/bin/splunk start` For additional details on how to configure forwarding with `outputs.conf` review this document

<http://docs.splunk.com/Documentation/Forwarder/latest/Forwarder/Configureforwardingwithoutputs.conf>

Universal Forwarder: Misconfiguration in `inputs.conf`

To specify what data the forwarder should collect, you must configure the inputs in the `inputs.conf` configuration file. The accuracy of the syntax and details saved in this file are critical to collecting data off the Linux server.

If for example you're using the forwarder to watch all files in a path or a single file, you must specify the input type and then the path, so ensure that you put three slashes in the path if the path includes the root directory.

1. On your forwarder, open

```
$SPLUNK_HOME/etc/system/local/inputs.conf.
```

This is an example syntax if you're monitoring everything in `/apache/foo/logs` or `/apache/bar/logs`, etc.

```
[monitor:///apache/.../logs]
index = web
sourcetype = access_combined
```

.....
This is another example if you're monitoring everything in `/apache/` that ends in `.log` (note that you can use wildcards for the path).

```
[monitor:///apache/*.log]
```

```
index = prod
sourcetype = apache
```

For additional details on monitoring files and directories with inputs.conf review this document

<https://docs.splunk.com/Documentation/Splunk/latest/Data/Monitorfilesanddirectorieswithinputs.conf>