

Vandalay Industries Monitoring Activity

Let's Go Splunking!

Task 1: Create a report to determine the impact that the DDOS attack had on download and upload speed. Additionally, create an additional field to calculate the ratio of the upload speed to the download speed.

Server_Speedtest.csv was uploaded and loaded into Splunk Enterprise with the source type set to .csv for proper formatting.

Set Source Type

This page lets you see how the Splunk platform sees your data before indexing. If the events look correct and have the right timestamps, click "Next" to proceed. If not, use the options below to define proper event breaks and timestamps. If you cannot find an appropriate source type for your data, create a new one by clicking "Save As".

Source: **server_speedtest.csv**

Source type: **csv** Save As

	_time	CONNECTION_MODE	DISTANCE_MILES	DOWNLOAD_MEGABITS	IP_ADDRESS	LATENCY
1	2/20/20 2:21:00.000 PM	multi	0	109.16	198.153.194.1	12
2	2/21/20 2:30:00.000 PM	multi	0	105.91	198.153.194.1	10
3	2/21/20 4:30:00.000 PM	multi	1	106.91	198.153.194.2	11
4	2/21/20 6:30:00.000 PM	multi	2	107.91	198.153.194.2	12

Review

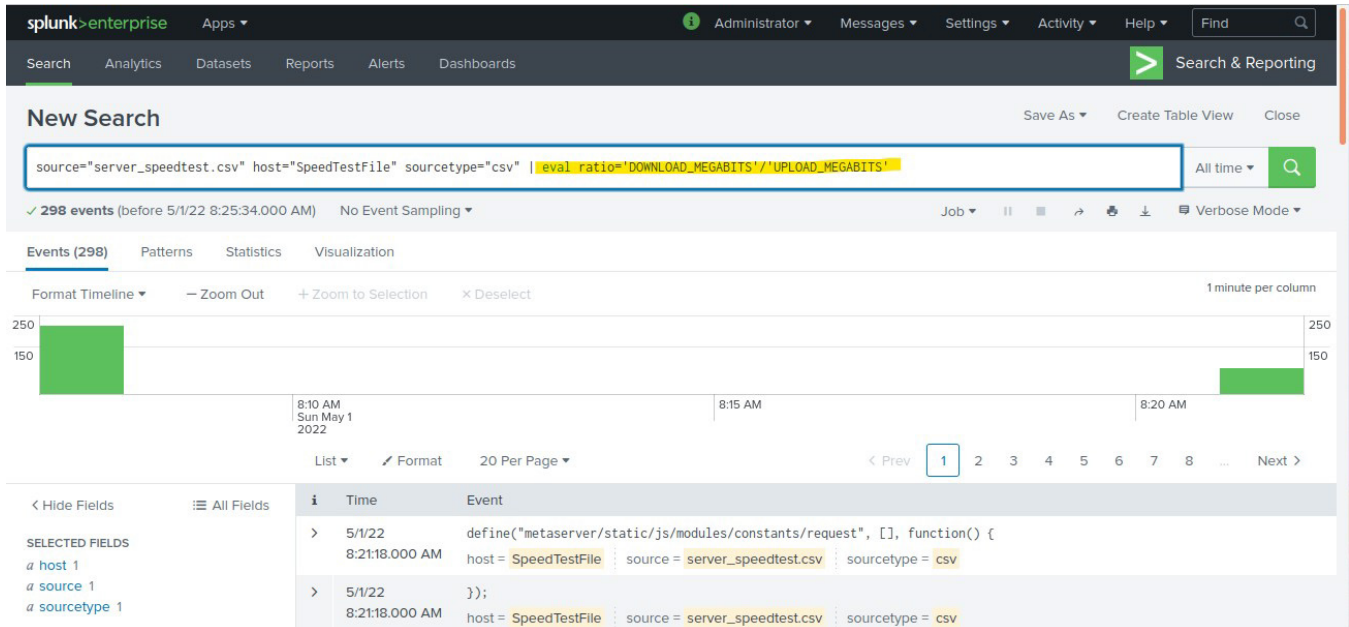
Input Type Uploaded File
 File Name server_speedtest.csv
 Source Type csv
 Host ServerSpeedTest
 Index Default

After successfully loading the log file into Splunk, I begin with some search SPL queries related to speeds and bandwidth over time to determine any spikes or possibly abnormal traffic.

Frank Lin - Unit 18 Homework - Let's Go Splunking!

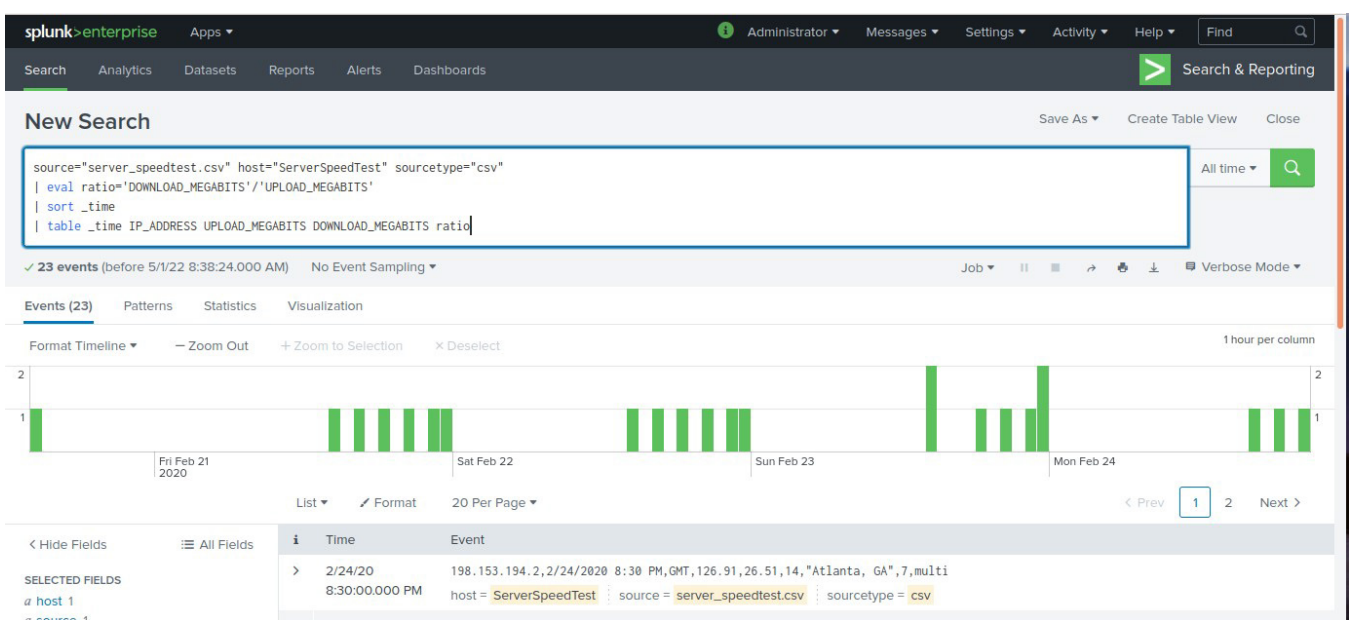
I use the `eval` command and create a new `ratio` field that shows the ratio between the upload and download speeds.

```
source="server_speedtest.csv" host="ServerSpeedTest" sourcetype="csv"
| eval ratio='DOWNLOAD_MEGABITS'/'UPLOAD_MEGABITS'
```



In order to sort the data so that it can be evaluated better, I use the `table` command to create a statistics table with the desired fields and new ratio calculation of upload vs download speeds.

```
source="server_speedtest.csv" host="ServerSpeedTest" sourcetype="csv"
| eval ratio='DOWNLOAD_MEGABITS'/'UPLOAD_MEGABITS'
| sort _time
| table _time IP_ADDRESS UPLOAD_MEGABITS DOWNLOAD_MEGABITS ratio
```



Frank Lin - Unit 18 Homework - Let's Go Splunking!

The results of those findings are as follows:

localhost:8000/en-US/app/search/search?earliest=0&latest=&q=search source%3D"server_speedtest.csv" host%3D

```

source="server_speedtest.csv" host="ServerSpeedTest" sourcetype="csv"
| eval ratio='DOWNLOAD_MEGABITS'/'UPLOAD_MEGABITS'
| sort _time
| table _time IP_ADDRESS UPLOAD_MEGABITS DOWNLOAD_MEGABITS ratio
    
```

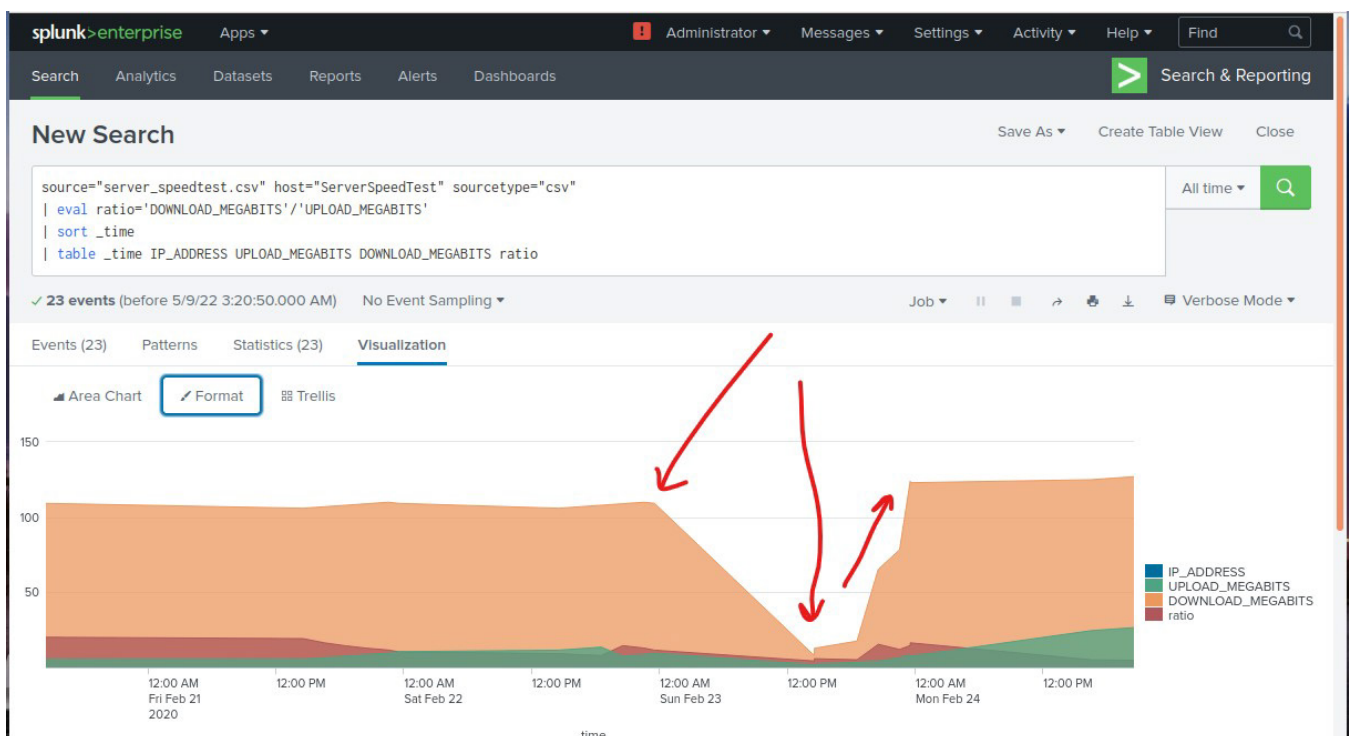
✓ 23 events (before 5/1/22 7:30:51.000 PM) No Event Sampling

Events (23) Patterns Statistics (23) Visualization

20 Per Page Format Preview

_time	IP_ADDRESS	UPLOAD_MEGABITS	DOWNLOAD_MEGABITS	ratio
2020-02-20 14:21:00	198.153.194.1	5.43	109.16	20.1
2020-02-21 14:30:00	198.153.194.1	5.51	105.91	19.2
2020-02-21 16:30:00	198.153.194.2	6.51	106.91	16.4
2020-02-21 18:30:00	198.153.194.2	7.51	107.91	14.4
2020-02-21 20:30:00	198.153.194.1	8.51	108.91	12.8
2020-02-21 22:30:00	198.153.194.1	9.51	109.91	11.6
2020-02-21 23:30:00	198.153.194.1	10.51	109.16	10.39
2020-02-22 14:30:00	198.153.194.1	11.51	105.91	9.202
2020-02-22 16:30:00	198.153.194.2	12.51	106.91	8.546
2020-02-22 18:30:00	198.153.194.2	13.51	107.91	7.987
2020-02-22 20:30:00	198.153.194.2	7.51	108.91	14.5
2020-02-22 22:30:00	198.153.194.2	8.51	109.91	12.9
2020-02-22 23:30:00	198.153.194.2	9.51	109.16	11.5
2020-02-23 14:30:00	198.153.194.1	1.83	7.87	4.30
2020-02-23 14:30:00	198.153.194.2	2.19	12.76	5.83
2020-02-23 18:30:00	198.153.194.2	3.43	17.56	5.12
2020-02-23 20:30:00	198.153.194.2	4.23	65.34	15.4
2020-02-23 22:30:00	198.153.194.1	6.51	78.34	12.0
2020-02-23 23:30:00	198.153.194.2	8.51	121.91	14.6

The findings of this search query shows that the internet speeds were affected on February 23, 2020 at around 14:30. It wasn't until about 20:30 when the systems started to recover from the attack, returning to normal at around 23:30. The duration from 14:30 to 23:30 indicates that the systems took almost 9 hours to recover from the attack.



Task 2: Create a report determining how many critical vulnerabilities exist on the customer data server. Then, build an alert to notify your team if a critical vulnerability reappears on this server.

The nessus_logs.csv was uploaded and loaded into Splunk Enterprise with the source type set to .csv for proper formatting. After successfully loading the logs to review, I did a search of any critical vulnerabilities that exist on the customer database server with the following SPL query:

```
source="nessus_logs.csv" host="NessusScanResults" sourcetype="csv"
severity=critical
| stats count by severity
```

The screenshot shows the Splunk Enterprise interface. At the top, there's a navigation bar with 'splunk>enterprise' and various menu items like 'Apps', 'Administrator', 'Messages', 'Settings', 'Activity', 'Help', and a 'Find' search bar. Below this is a 'Search & Reporting' section with tabs for 'Search', 'Analytics', 'Datasets', 'Reports', 'Alerts', and 'Dashboards'. The 'Search' tab is active, showing a 'New Search' window. The search query is: `source="nessus_logs.csv" host="NessusScanResults" sourcetype="csv" dest_ip="10.11.36.23" severity=critical | stats count by severity`. The results show 98 events (before 5/9/22 7:15:55.000 AM) with 'No Event Sampling'. The 'Statistics (1)' tab is selected, showing a table with one row: 'critical' with a count of 98.

98 critical vulnerabilities were discovered. Next was to build an alert that monitors everyday to see if this server has any new critical vulnerabilities and to notify the SOC via email right away. See below.

The screenshot shows the 'Save As Alert' dialog box in Splunk Enterprise. The dialog has a 'Settings' section with fields for 'Title' (Critical Alert - Nessus Database Server), 'Description' (Database IP 10.11.36.23), 'Permissions' (Private), 'Alert type' (Scheduled), and 'Run every day'. The 'Trigger Conditions' section shows 'Trigger alert when' set to 'Number of Results' (is greater than 0) and 'Trigger' set to 'Once'. There are 'Cancel' and 'Save' buttons at the bottom.

Frank Lin - Unit 18 Homework - Let's Go Splunking!

Save As Alert

When triggered: Send email Remove

To:

Comma separated list of email addresses. [Show CC and BCC](#)

Priority: Highest

Subject:

The email subject, recipients and message can include tokens that insert text based on the results of the search. [Learn More](#)

Message:

Include:

- ☒ Link to Alert
- ☒ Link to Results
- ☐ Search String
- ☐ Inline Table
- ☐ Trigger
- ☐ Attach CSV
- ☐ Condition
- ☐ Trigger Time
- ☐ Attach PDF

Cancel Save

Critical Alert - Nessus Database Server Edit

Database IP 10.11.36.23

Enabled: Yes. [Disable](#)

App: search

Permissions: Private. Owned by admin. [Edit](#)

Modified: May 9, 2022 7:20:54 AM

Alert Type: Scheduled, Daily, at 0:00. [Edit](#)

Trigger Condition: .. Number of Results is > 0. [Edit](#)

Actions: 1 Action [Edit](#)

- Send email

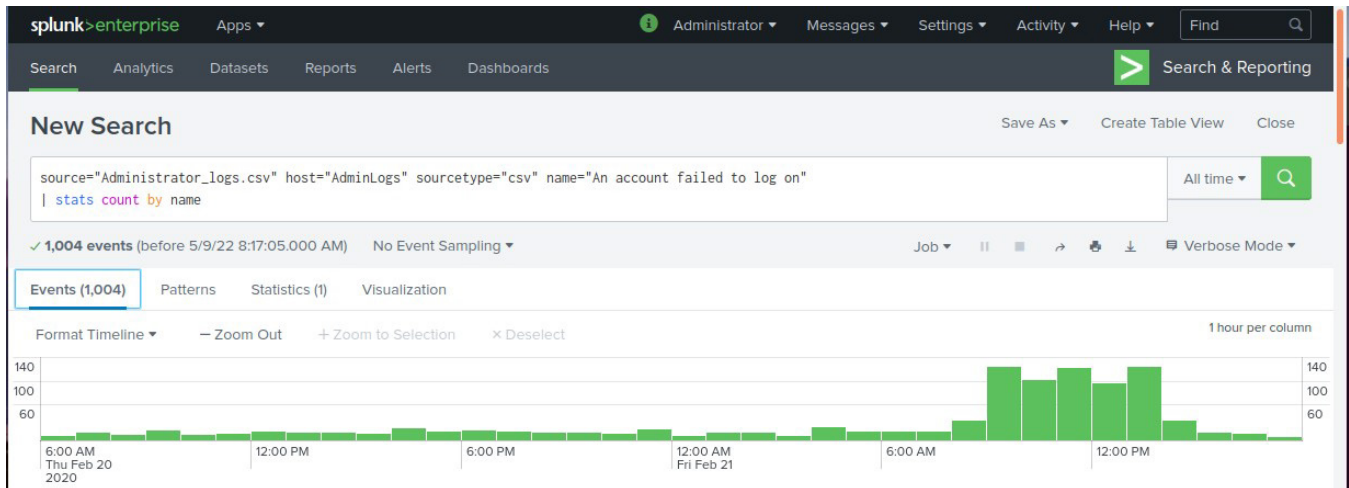
i There are no fired events for this alert.

Task 3: Analyze administrator logs that document a brute force attack. Then, create a baseline of the ordinary amount of administrator bad logins and determine a threshold to indicate if a brute force attack is occurring.

The Administrator_logs.csv was uploaded and loaded into Splunk Enterprise with the source type set to .csv for proper formatting. After successfully loading the logs to review, I did a search for the brute force attack to see when it had occurred.

```
source="Administrator_logs.csv" host="AdminLogs" sourcetype="csv"
name="An account failed to log on"
```


Frank Lin - Unit 18 Homework - Let's Go Splunking!



By examining the information gathered from the logs, it appears that the attack occurred around 9:00am (some activity may have started a little before 9:00am) and ended around 2:00pm on February 21, 2020. The total timeframe of the attack was about 5 hours.

Based on the timeline view of the logs, the baseline of normal activity for failed log ons should be about up to 34 per hour. From the looks of things, I have decided that it is worth alerting for a brute force attack if about 50 failed log ons occur per hour.

The screenshot shows the Splunk interface with the same search query. The results are displayed in a table view. The table has two columns: 'name' and 'count'. The 'name' column contains the text 'An account failed to log on'. The 'count' column shows the value 1004.

name	count
An account failed to log on	1004

A stats query to sort the failed log on attempts count was created so that I can make an alert based on the event inquiry if there is an abnormal activity count based on the category of log events that contain "An account failed to log on."

Since the SOC team should be notified as soon as possible, an alert was set up to check every hour to see if that threshold determined is surpassed during any hour that would result in a potential brute force attack happening so that they can look into it further and take action.

The alert information screenshots are on the following pages.

The image displays two screenshots of the Splunk 'Save As Alert' dialog box, overlaid on a background of the Splunk interface showing a search for 'Administrator_logs'.

Top Screenshot: Initial Alert Configuration

- Title:** Brute Force Attempt - High Number of Failed Log Ons
- Description:** Optional
- Permissions:** Private
- Alert type:** Scheduled
- Run every:** Run every hour
- At:** 0 minutes past the hour
- Expires:** 30 day(s)
- Trigger Conditions:**
 - Trigger alert when: Number of Results
 - Is greater than: 50
 - Trigger: Once
 - Throttle: ☐
- Trigger Actions:** (Empty)

Bottom Screenshot: Email Notification Configuration

- When triggered:** Send email
- To:** soc@vandalay.com
- Priority:** Highest
- Subject:** Brute Force Attack!!!!
- Message:** The alert condition for '\$name\$' was triggered. Potential Brute Force Attack happening!!
- Include:**
 - ☒ Link to Alert
 - ☒ Link to Results
 - ☐ Search String
 - ☐ Inline Table
 - ☐ Trigger Condition
 - ☐ Attach CSV
 - ☐ Trigger Time
 - ☐ Attach PDF

Frank Lin - Unit 18 Homework - Let's Go Splunking!

The screenshot shows the Splunk Enterprise web interface. The top navigation bar includes the Splunk logo, 'enterprise' version, and user 'Administrator'. Below the navigation bar, the 'Alerts' tab is selected. The main content area displays the configuration for an alert titled 'Brute Force Attempt - High Number of Failed Log Ons'. The configuration includes fields for 'Enabled' (Yes), 'App' (search), 'Permissions' (Private), 'Modified' (May 9, 2022 8:25:07 AM), and 'Alert Type' (Scheduled). The 'Trigger Condition' is set to 'Number of Results is > 50'. The 'Actions' section shows '1 Action' with a 'Send email' option. A message at the bottom states 'There are no fired events for this alert.'

Brute Force Attempt - High Number of Failed Log Ons Edit

Enabled: Yes. [Disable](#) Trigger Condition: .. Number of Results is > 50. [Edit](#)

App: search Actions: [1 Action](#) [Edit](#)

Permissions: Private. Owned by admin. [Edit](#) [Send email](#)

Modified: May 9, 2022 8:25:07 AM

Alert Type: Scheduled. Hourly, at 0 minutes past the hour. [Edit](#)

i There are no fired events for this alert.

Now the SOC team can watch more closely at the administrator account without having to watch it all day long. Another time-saving action that I was able to help contribute for the good people at Vandalay Industries. Gotta love Splunking!