## **Jacob Schoellkopf**

## **Let's go Splunking!**

### **My Scenario**

I have just been hired as an SOC Analyst by Vandalay Industries, an importing and exporting company.

* Vandalay Industries uses Splunk for their security monitoring and have been experiencing a variety of security issues against their online systems over the past few months.
* You are tasked with developing searches, custom reports and alerts to monitor Vandalay's security environment in order to protect them from future attacks.

### **Tools Used from My Home Lab**

I am using the Splunk app located in my Ubuntu Virtual Machine.

### **My Objective**

I am using my knowledge of SIEM to design a powerful monitoring solution to protect Vandaly from security attacks.

### **Topics I am Covering:**

* Researching and adding new apps
* Installing new apps
* Uploading files
* Splunk searching
* Using fields
* Custom reports
* Custom alerts

## **Vandalay Industries Monitoring Actiavity Instructions**

### **Step 1: The Need for Speed**

**Background**: As the worldwide leader of importing and exporting, Vandalay Industries has been the target of many adversaries attempting to disrupt their online business. Recently, Vandalay has been experiencing DDOS attacks against their web servers.

Not only were web servers taken offline by a DDOS attack, but upload and download speed were also significantly impacted after the outage. My networking team has provided the results of a network speed run around the time of the latest DDOS attack.

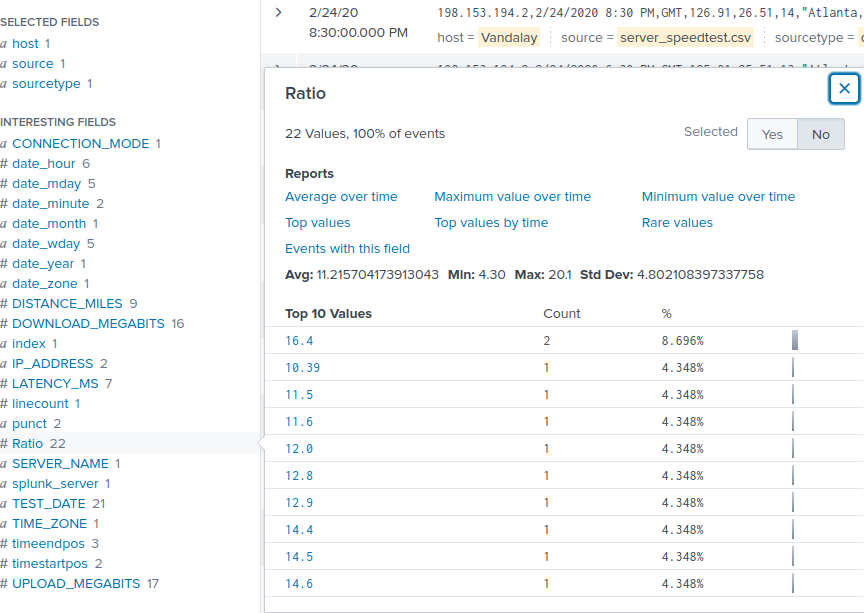
**My Task:** Is to create a report to determine the impact that the DDOS attack had on download and upload speed. Additionally, I am creating a field to calculate the ratio of the upload speed to the download speed.

First, I am uploading the following file of the system speeds around the time of the attack.



Second, using the eval command, I created a field called ratio that shows the ratio between the upload and download speeds.

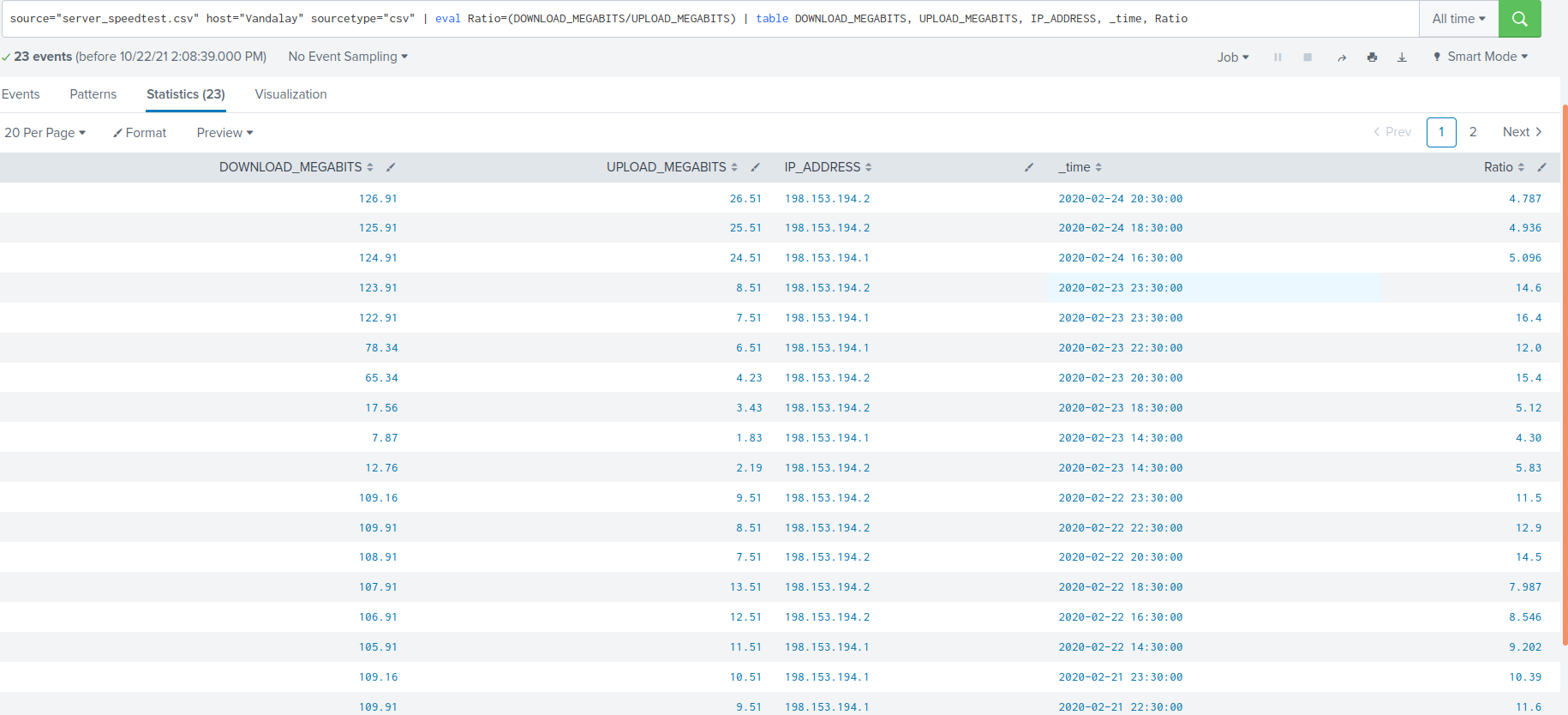
* + Command used: **| eval ratio=(DOWNLOAD\_MEGABITS/UPLOAD\_MEGABITS)**

My Image: 

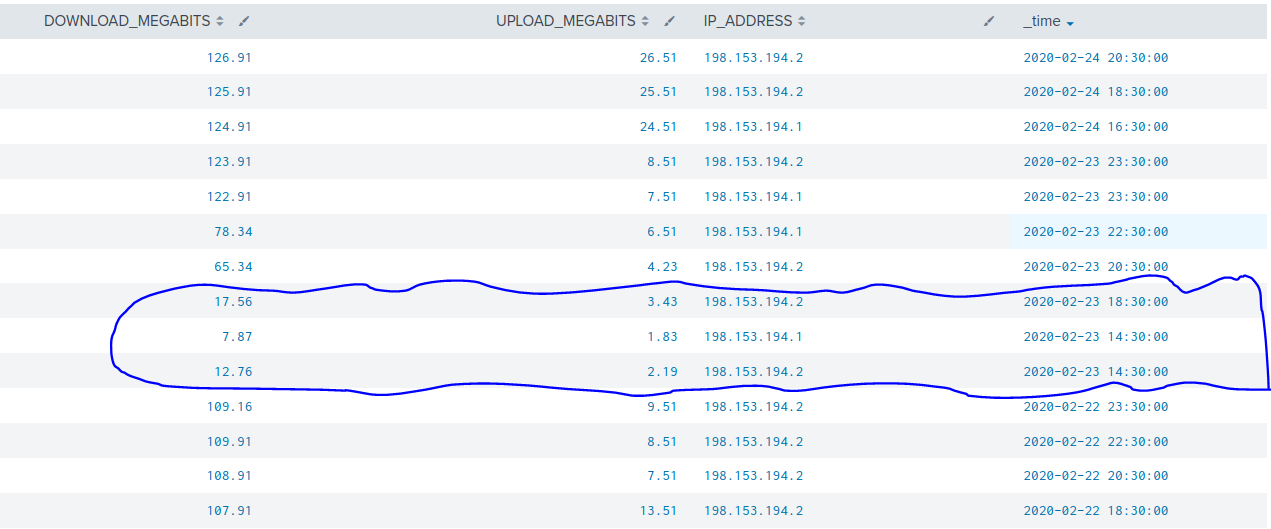
Under top 10 values shows the ratios of Download\_megabits to Upload\_megbits.

Third, I created a report using the Splunk's table command to display the following fields in a statistics report:

* + \_time
  + IP\_ADDRESS
  + DOWNLOAD\_MEGABITS
  + UPLOAD\_MEGABITS
  + ratio



The command I used is: **| table DOWNLOAD\_MEAGABITS, UPLOAD\_MEGABITS, IP\_ADDRESS, \_time, ratio**

Next, based on the report created, I searched for the approximate date and time of the attack. I arranged the time column to show me the events in reverse chronological order. The image below displays the following:

As illustrated above, the downloaded megabits from 2:30 pm on February 23rd to 6:30 pm on February 23rd had significantly less downloaded megabits, and this is most likely the timeframe that the attack occurred.

Next, I identified how long it took for my systems to recover.As seen above, the downloaded megabits started picking back up at 8:30 pm on February 23rd but did not fully recover until 11:30 pm on February 23rd. The last time the systems were functioning properly was on February 22nd at 11:30 pm. This means it took 24 hours for the system to fully recover. That is unacceptable and while this may be an unpopular opinion, the SOC team needs to start operating at a 24-hour schedule in order to detect and prevent similar attacks such as the one being illustrated from occurring.

### 

### **Step 2: So, Are We Vulnerable?**

**Background:** Due to the frequency of attacks, the manager needs to be sure that sensitive customer data on their servers is not vulnerable. Since Vandalay uses Nessus vulnerability scanners, I have pulled the last 24 hours of scans to see if there are any critical vulnerabilities.

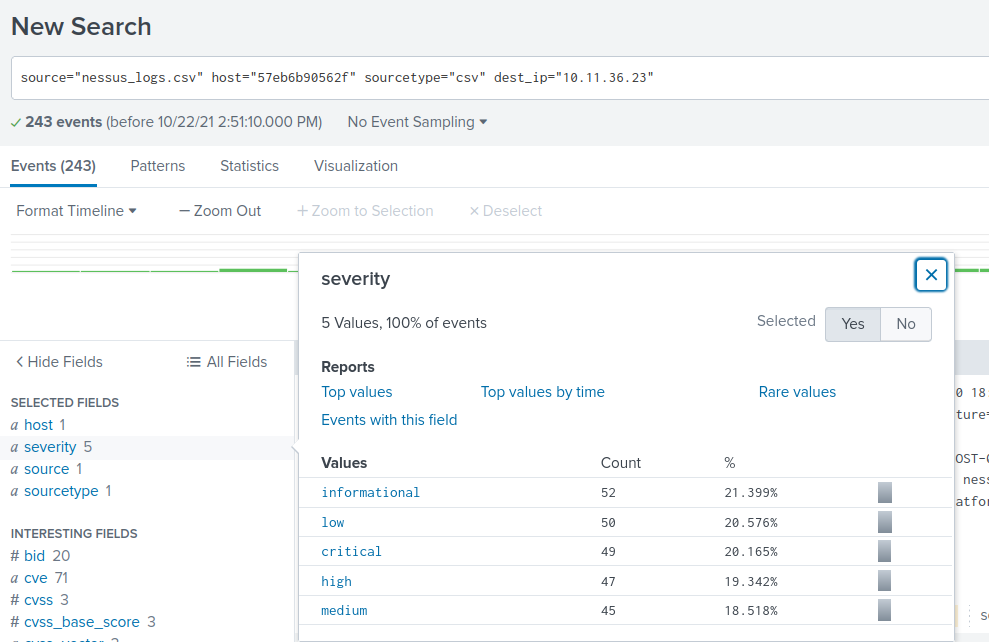
**My Task:** I will create a report determining how many critical vulnerabilities exist on the customer data server. Then, build an alert to notify your team if a critical vulnerability reappears on this server.

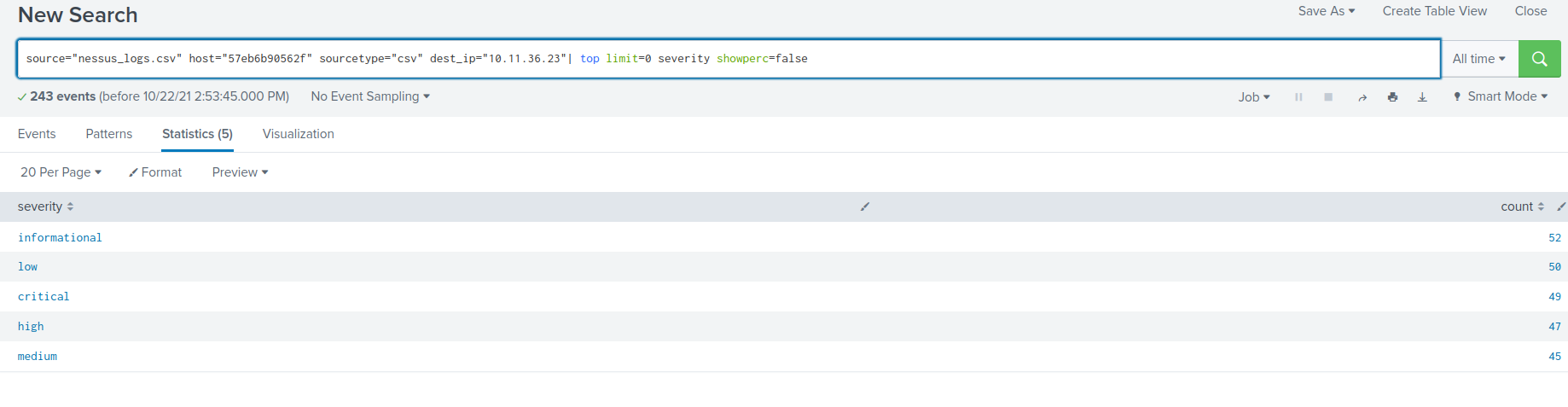
First, I uploaed the following file from the Nessus vulnerability scan.



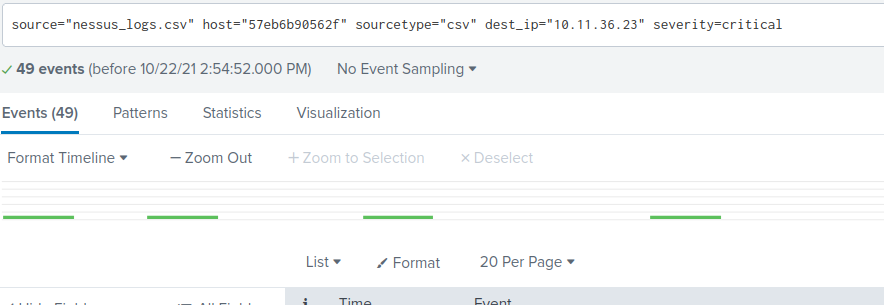
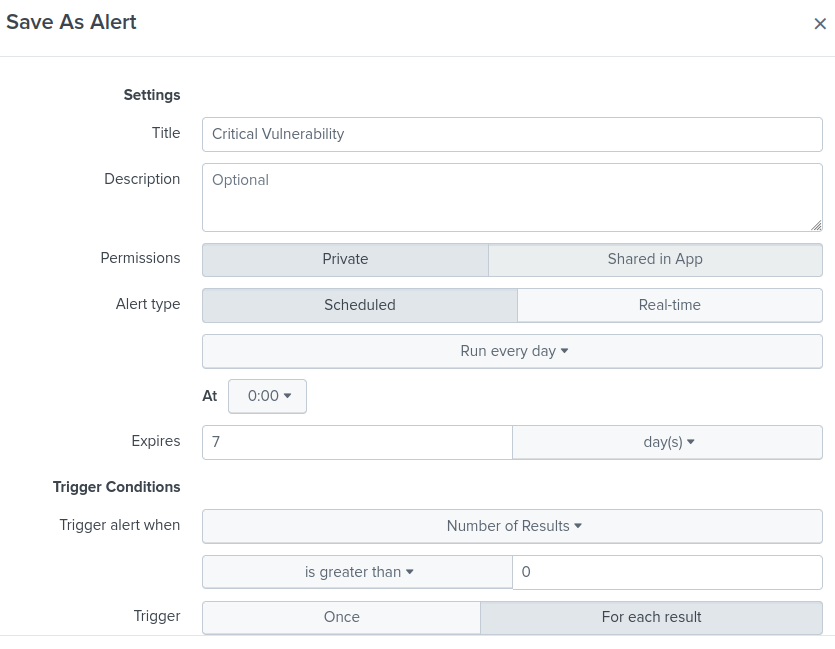
Next: I created a report that shows the count of critical vulnerabilities from the customer database server.

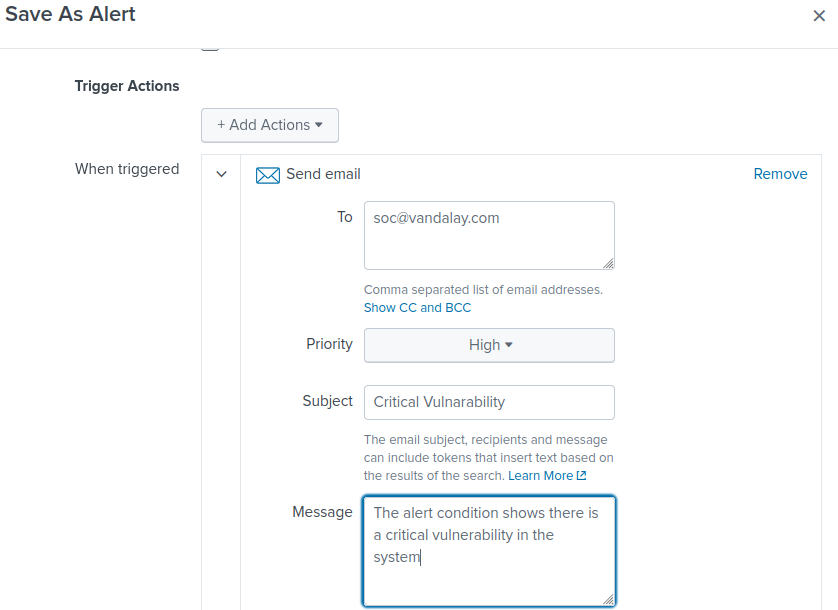
* + The database server IP is 10.11.36.23.
  + The field that identifies the level of vulnerabilities is severity.





Then, I built an alert that monitors every day to see if this server has any critical vulnerabilities. If a vulnerability does exist, I have an alert emailed to soc@vandalay.com.



**Step 3: Drawing the (base)line**

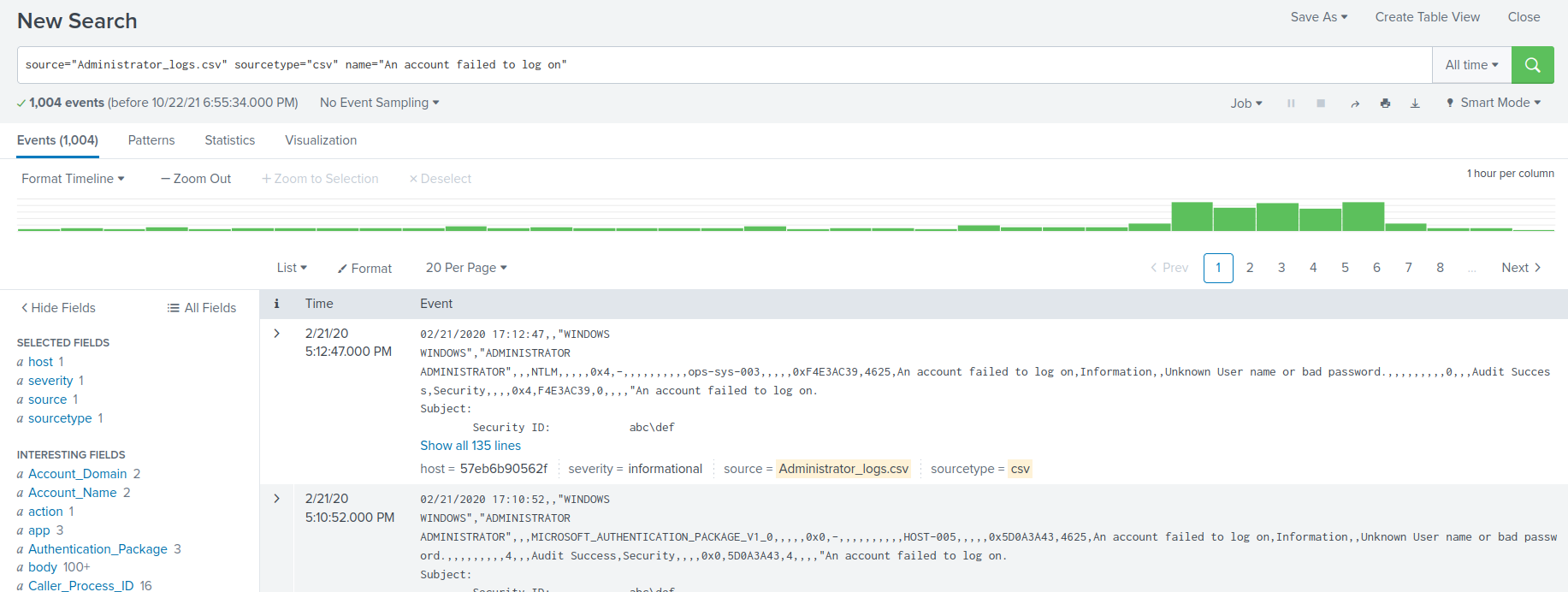
**Background:** A Vandalay server is also experiencing brute force attacks into their administrator account. Management would like to set up monitoring to notify the SOC team if a brute force attack occurs again.

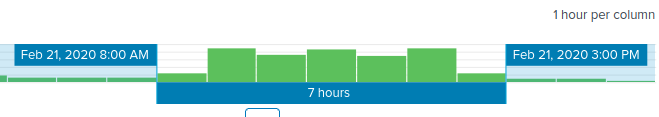
**My Task:** I need to analyze the logs that document a brute force attack. Then, I have to create a baseline of the ordinary amount of administrator bad logins and determine a threshold to indicate if a brute force attack is occurring.

First, I upload the administrator login logs.

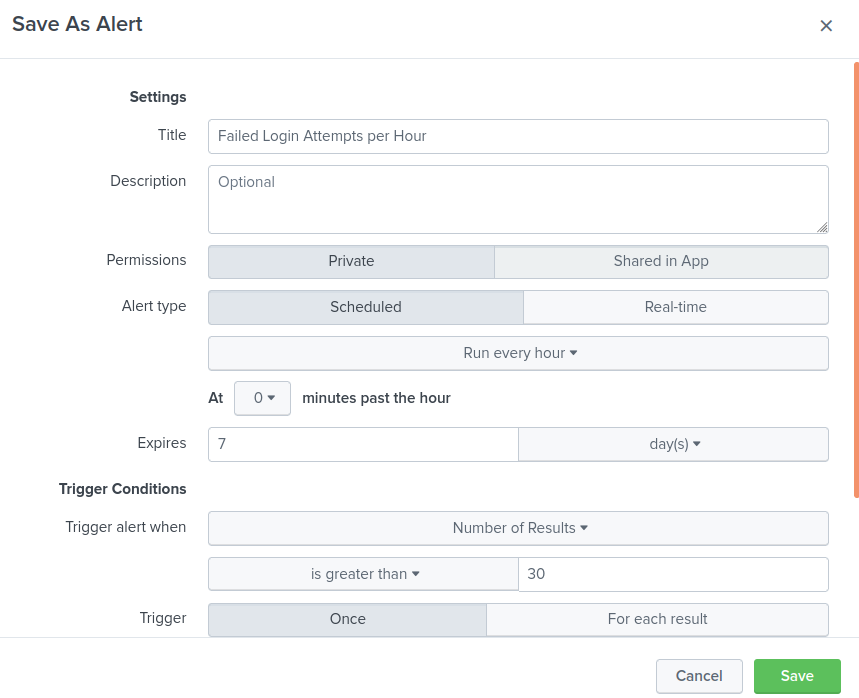


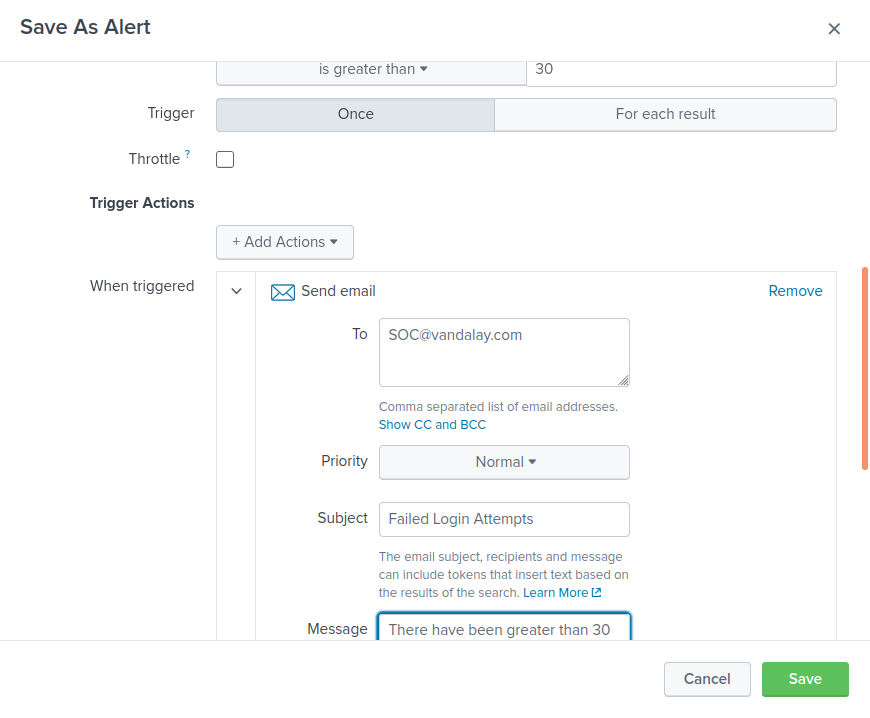
Second, I need to identify when the brute force attack occurred.





Analysis: This attack lasted for 7 hours. This department needs use a maximum of password attempts, blocks IP addresses that attempt to log on more than these attempts and utilize multi-factor. Also, since brute force attacks are web-based attacks, we need to make sure that we have strong input validation. This could prevent other attacks such as XSS or SQL injections.

Third: I need to determine a baseline of normal activity and a threshold that would alert if a brute force attack was occurring.  


Finally: I designed an alert to check the threshold every hour and email the SOC team at SOC@vandalay.com if triggered.  


With the newly implemented custom reports and alerts to monitor and harden Vandalay's network, Vandalay has a more secure environment that may protect future attacks from occuring.