



Splunk Reports and Alerts

Cybersecurity
SIEM Day 3



Class Objectives

By the end of class, you will be able to:



Use the SPL commands **stats** and **eval** to create new fields in Splunk.



Schedule statistical reports in Splunk.



Determine baselines of normal activity to trigger alerts.



Design and schedule alerts to notify if an attack occurs.



Activity: Splunk Warm Up

In this activity, you will analyze logs from a Fortinet IPS system, determine the security issue, and provide mitigation strategies.

Suggested Time:

0:10





Time's Up! Let's Review.

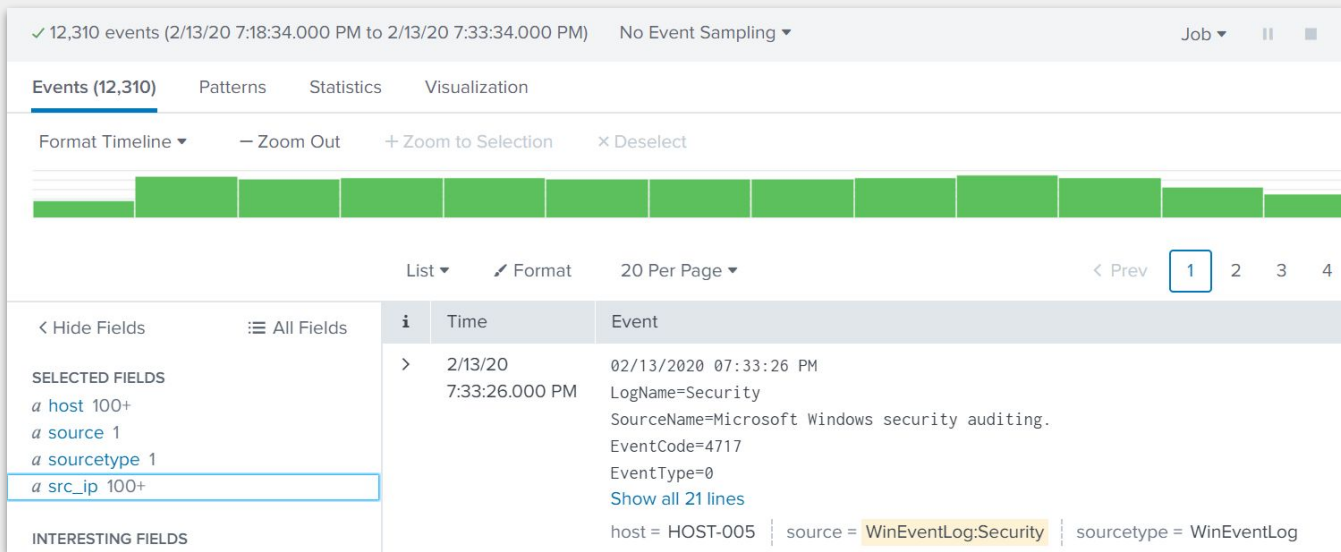
splunk[®] > **statistics**

Splunk Statistics

Security professional often need to present Splunk search results to non-technical audiences using simple formats.

For example

If we need to illustrate the top 10 IP addresses from a DOS attack, this results page could be confusing to a non-expert.



Splunk Statistics

Splunk uses the Statistics feature to display specific data points from search results in an easy-to-read format.

The **stats** command is the most basic Splunk command to create a statistics report.

The screenshot displays the Splunk Statistics interface. On the left, the 'INTERESTING FIELDS' list includes 'Account_Name' with 11 values. The main panel shows the 'Account_Name' field selected, with 11 values and 80% of events. The 'Reports' section is active, showing the 'Top 10 Values' report. The table below lists the top 10 values for 'Account_Name'.

Top 10 Values	Count	%
user_n	2	16.667%
ADMINISTRATOR ADMINISTRATOR	1	8.333%
BUSDEV-008 user_m	1	8.333%
PROD-POS-003 user_c	1	8.333%
user_a	1	8.333%
user_b user_d	1	8.333%
user_d	1	8.333%
user_d user_g	1	8.333%
user_e user_i	1	8.333%
user_f	1	8.333%



In the following demonstration, we will use the `stats` command to create a statistical report of the top `Account_Name` being targeted in a brute force attack.



Instructor Demonstration

Splunk Statistics

Creating New Fields

For example, suppose we are analyzing logs for potential brute force attempts. We can use Splunk to create new fields and display their statistics.

Our logs have event codes (**EventCode**) that assign numerical codes to events.

EventCode 4740 indicates a user logout.

Since user lockouts are a potential identifier of a brute force attack, we can create a new field to identify events that contain this event code.

If the **EventCode** field has a value of **4740**, the field value will be **Potential Brute Force**.

If the field does not have a value of **4740**, the field value will be **Not Brute Force**.

The eval Command

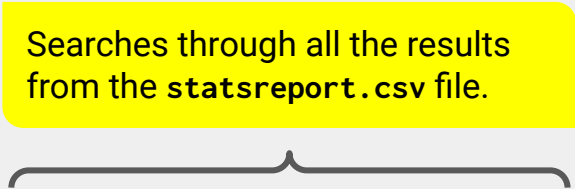
We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

The eval Command

We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

Searches through all the results
from the `statsreport.csv` file.




```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

The eval Command

We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

Creates a new field called **BruteForce**.



```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

The eval Command

We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

States the following expression: "If the `EventCode` field has a value of `4740`."

The eval Command

We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

Continue the statement, "If true, name this value **Potential Brute Force**."

The eval Command

We can use the `eval` command expressions, such as `if`, and place the resulting values into a search field.

```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

Continues the statement,
"If false, name this value
Not Brute Force."

The eval Command

We can use the eval command expressions, such as if, and places the resulting values into a search field.

Searches through all the results from the `statsreport.csv` file.

Creates a new field called `BruteForce`.

```
source="statsreport.csv" | eval BruteForce =  
if('EventCode'="4740", "Potential Brute Force", "Not Brute Force")
```

States the following expression: "If the `EventCode` field has a value of `4740`."

Continues the statement, "If true, name this value `Potential Brute Force`."

Continues the statement, "If false, name this value `Not Brute Force`."



Activity: Splunk Statistics

In this activity, you will create statistical reports to illustrate details about the DOS attack.

Suggested Time:
0:15





Time's Up! Let's Review.

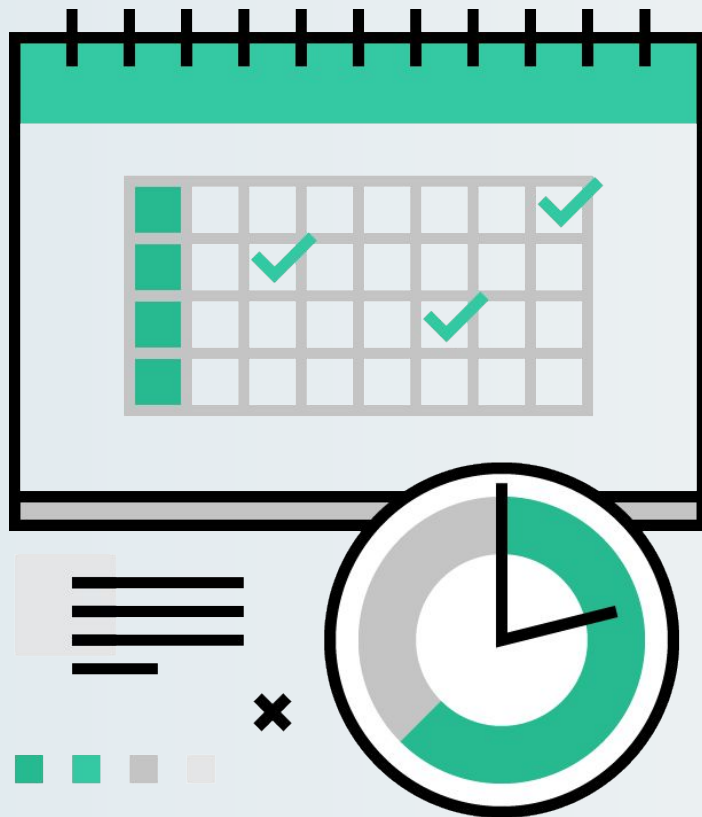
splunk[®] > **reports**

Splunk Reports

Statistical reports may need to be run at specific or reoccurring times.

For example, if an organization is experiencing suspicious network attacks around 12 a.m., they would want to analyze their network traffic every night at that time.

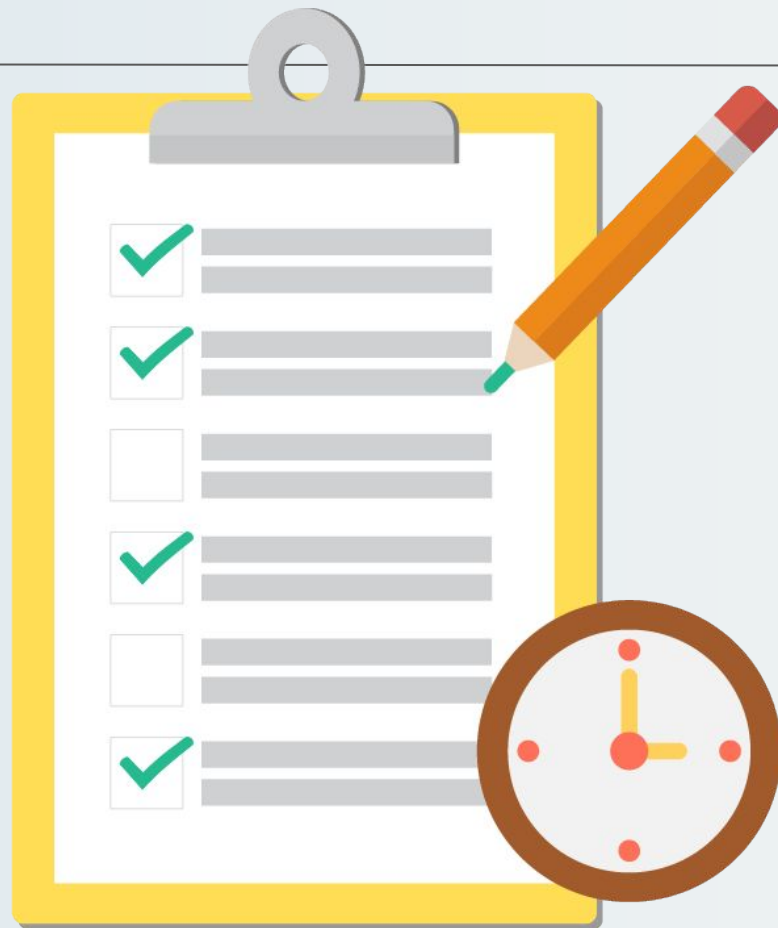
With Splunk, we can create and schedule custom reports to automate this task.



Splunk Report Demonstration

In the following demonstration, we will create and schedule a report using the continued scenario of monitoring brute force attacks:

- We were notified that the most recent brute force attacks happened around 12 a.m.
- Therefore, we will run a report at 1 a.m. each night to view activity for the past several hours.
- We'll also automate an email linking to the report after it runs.





Instructor Demonstration

Creating and Scheduling Reports



Activity: Splunk Reports

In this activity, you will schedule a statistical report for OMP management so they can review the current state of the attacks against a server.

Suggested Time:
0:07





Time's Up! Let's Review.



splunk[®] > **alerts**



Splunk alerts are designed to automatically notify the right people when a specific condition, known as a **trigger condition**, is met.

Splunk alerts are automatic.

Once they are created, Splunk's software checks the trigger condition.

- Splunk reports are scheduled.
- Splunk alerts are **triggered**.



Splunk Alerts

Trigger conditions contain the following criteria:

01

Search/Report Results

Indicate which criteria to check.

For example:
300 logins have been attempted.

02

Time parameters

Indicate the time period to check.

For example:
Within last 24 hours.

03

Schedule

Determines the frequency with which these criteria are checked.

For example:
Every day at 12 p.m.

Splunk Alerts



When the condition is met, a trigger action is executed to alert the Splunk user.
For example:

```
"Send an email to soc_manager@acme.com."
```

Baselining

Designing Strong Alerts

A required skill for designing strong alerts is being able to avoid **false positives** and **false negatives**.

	 False Positives	 False Negatives
What Occurred	Regular login activity	Brute force attack
Alerts	Yes	No
Outcome	Alerts went off but security professionals identified a non-issue.	No alerts went off, a brute-force attack occurred and several accounts were breached.

False Positives

False positives occur when conditions are met and an alert is triggered, but the security situation did not actually occur.

For example, an alert was created to detect suspicious login activity on our Linux server.

- The chosen criteria checks activity every hour and creates an alert when 10 logins occur within an hour.
- Several alerts were triggered, but further research determined they were set off by normal user activity.
- SOC realizes that 10 login attempts within an hour is not very suspicious.



False Negatives

False negatives occur when the condition is met and an alert isn't triggered, allowing the security event to go undetected.

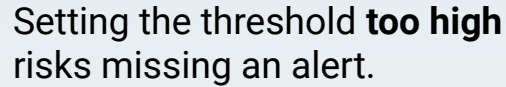
For example, an alert was created to detect suspicious activity of logins on our Linux server.

- The chosen criteria checks activity every hour and creates an alert when 500 logins occur within an hour.
- Suspicious login activity did occur on the server when an attacker tried to brute force the Linux server with 400 attempts, but no alerts were triggered.



Setting a Baseline Threshold

Baselining is a method of looking at historical data to determine normal activity, known as a **threshold**. When the threshold is exceeded, an alert is triggered.



Setting the threshold **too high**
risks missing an alert.

Setting the threshold **too low**
creates too many false positives.



Instructor Demonstration

Baselining

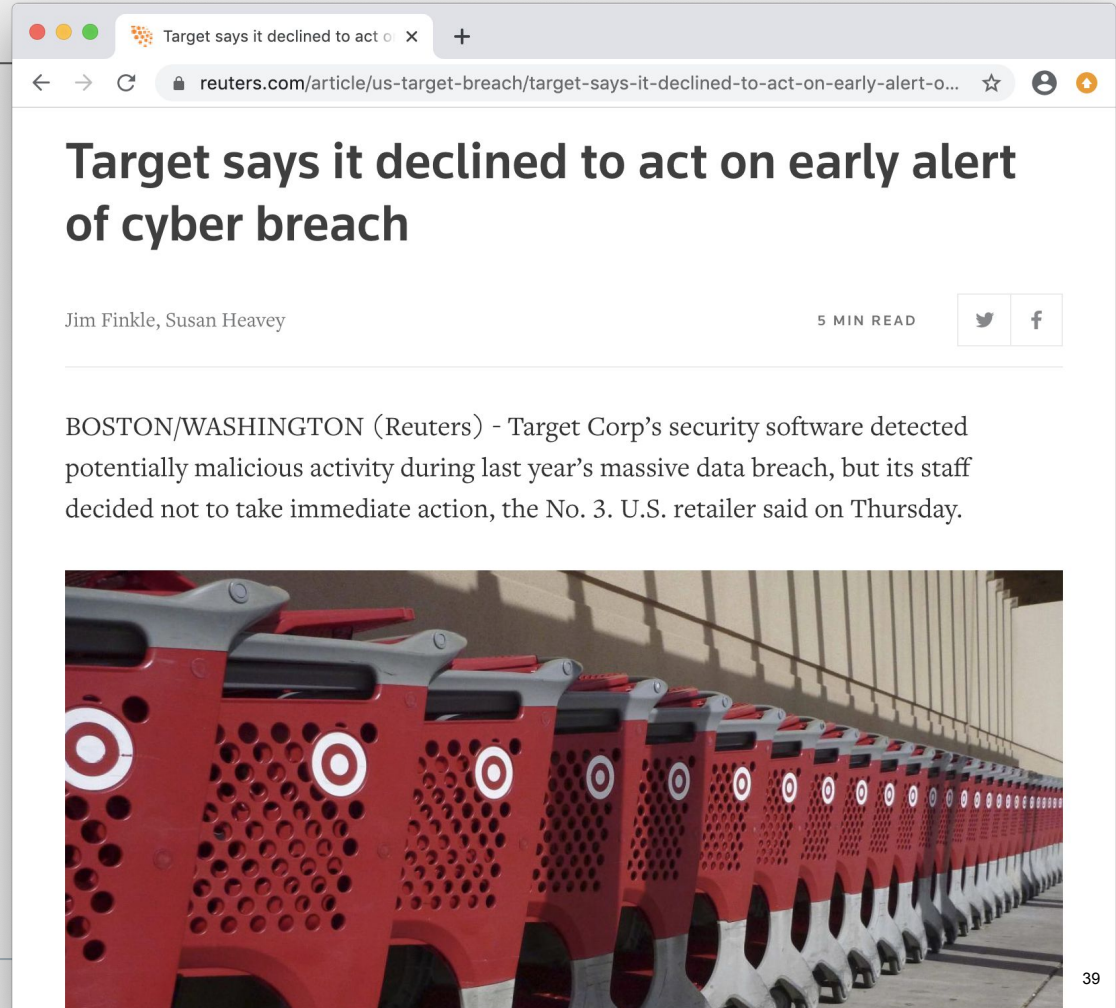
A woman with long brown hair and glasses, wearing a blue and white striped shirt and dark blue pants, is running to the right. She has a look of urgency or stress. Behind her is a large, dense, and chaotic cloud of various digital and communication icons. These icons include a smartphone, a calendar showing '13', a speech bubble with '1000', a clock, a magnifying glass, a heart, a question mark, a 'to do' list, a percentage sign, an '@' symbol, a location pin, a bar chart, a globe, a paper plane, a document, a speech bubble, a plus sign, a minus sign, a star, and a speech bubble with a checkmark. The background is a solid light yellow.

- Even when an organization builds good alerts and an alert gets triggered, security professionals will need to research and respond to it.
- If an organization's system triggers too many alerts, even if they are good alerts, security professionals will often miss issues as they get lost in the noise.

Alert Fatigue

Alert fatigue can have a major impact on organizations:

-  In 2014, a breach at Target cost the company \$252 million and led to the resignation of its CIO and CEO.
-  One of the company's security products actually detected the breach.
-  But due to the high quantity of alerts and the frequency of false alerts, the company's IT security team ignored it.





Activity: Baseline

In this activity, you will review logs and create a baseline of normal hourly login counts.

Suggested Time:
0:15





Time's Up! Let's Review.

Creating and Scheduling Alerts

Creating an Alert

Now that we can determine accurate baselines, we can continue with our scenario and design the alerts.

In the following demonstration, we will design an alert to trigger when 30 logins occur in an hour.

We will set this alert to check the count every hour.

Once the alert is triggered, an email will be sent.





Instructor Demonstration

Creating and Scheduling Alerts



Activity: Creating and Scheduling Alerts

In this activity, you will design and schedule an alert to notify your team if a brute force attack is occurring.

Suggested Time:
0:10





Time's Up! Let's Review.

*The
End*