SPLUNK: EXPLOITATION 2025/12/12

Exploitation Phase

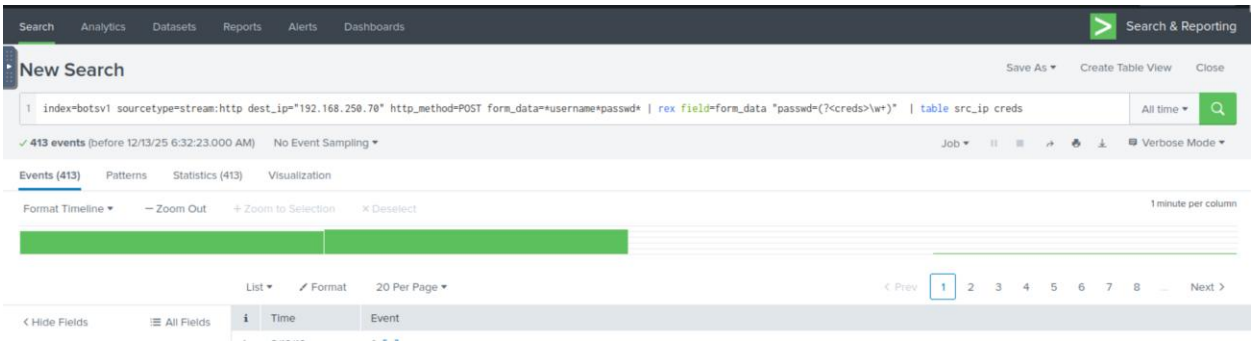
The attacker needs to exploit the vulnerability to gain access to the system/server.

In this task, we will look at the potential exploitation attempt from the attacker against our web server and see if the attacker got successful in exploiting or not.

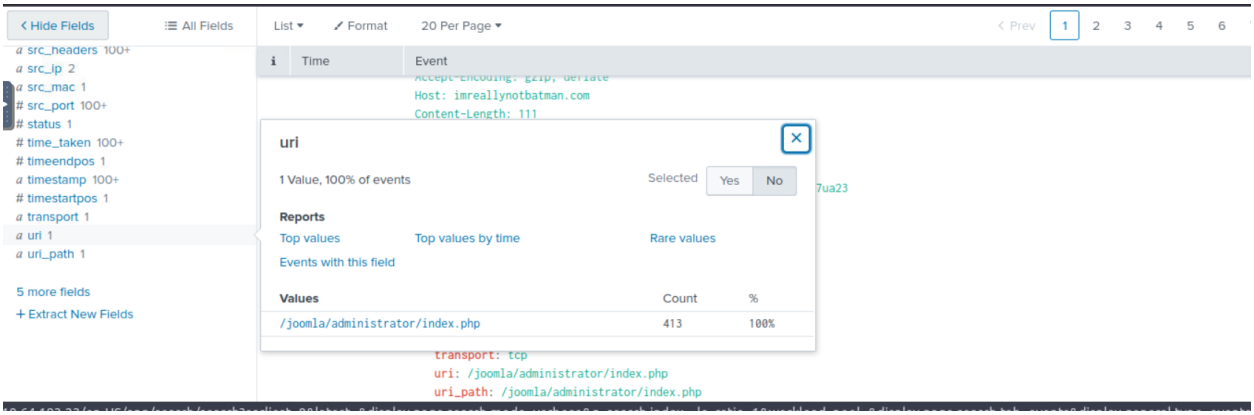
To begin our investigation, let's note the information we have so far:

- We found two IP addresses from the reconnaissance phase with sending requests to our server.
- One of the IPs 40.80.148.42 was seen attempting to scan the server with IP **192.168.250.70**.
- The attacker was using the web scanner Acunetix for the scanning attempt.

STEP 1: Finding the URI which got the multiple brute force attempts?

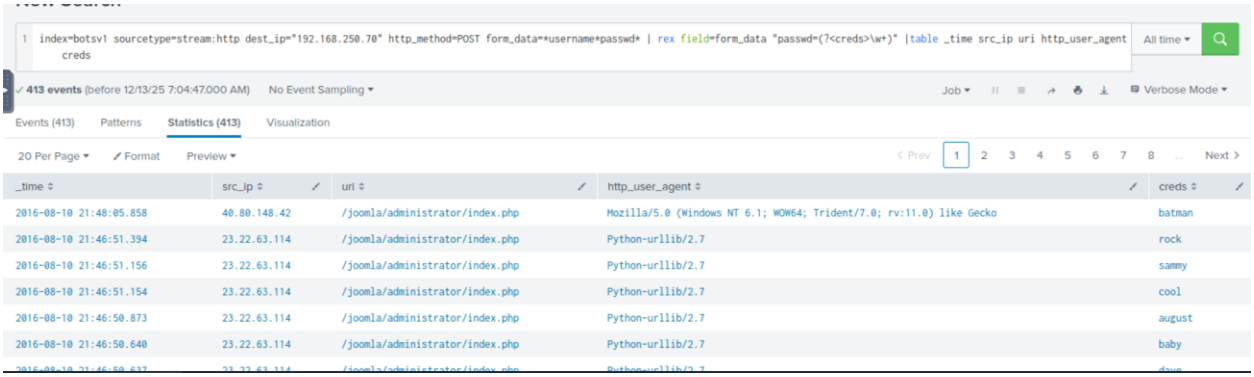


Since we're working with a brute-forcing attempt, I went right ahead and used the following search query '**index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST form_data=*username*passwd* | rex field=form_data "passwd=(?<creds>\w+)" | table src_ip creds**'. Since we were dealing with a brute-force attempt, I focused on POST requests containing form data, as login attempts typically transmit credentials this way. The following query was used to extract password submission attempts and identify the source IPs involved. This helps confirm brute-force behaviour by highlighting repeated credential activity.

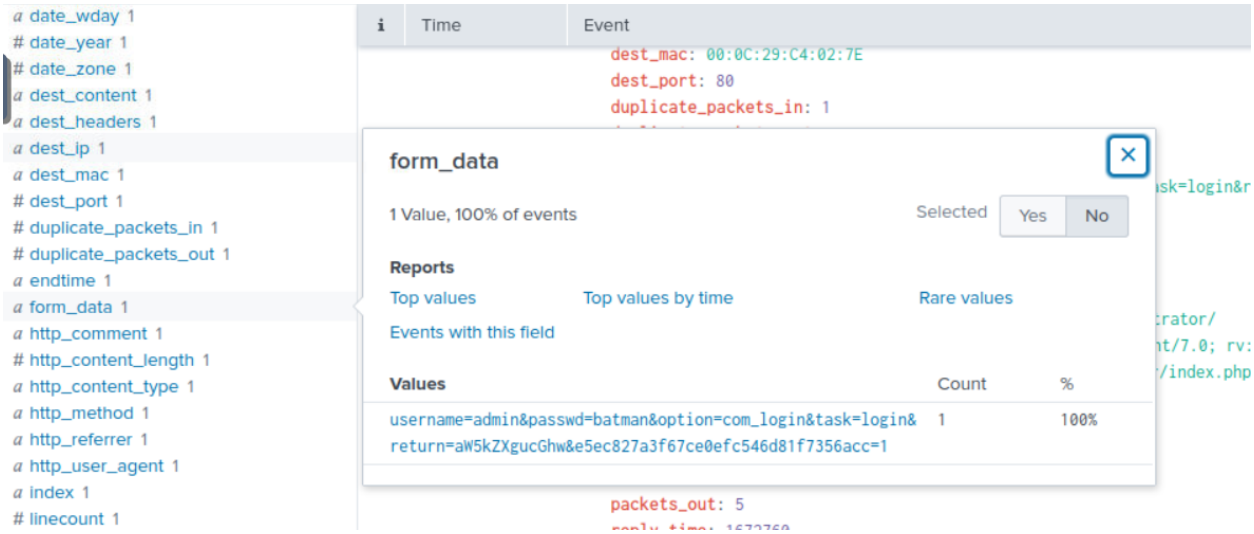


I went to the left and scrolled down the fields to find the uri. I opened it and saw that the value '**/joomla/administrator/index.php**' was shown. I went with it and the answer was correct.

STEP 2: Against which username was the brute force attempt made?

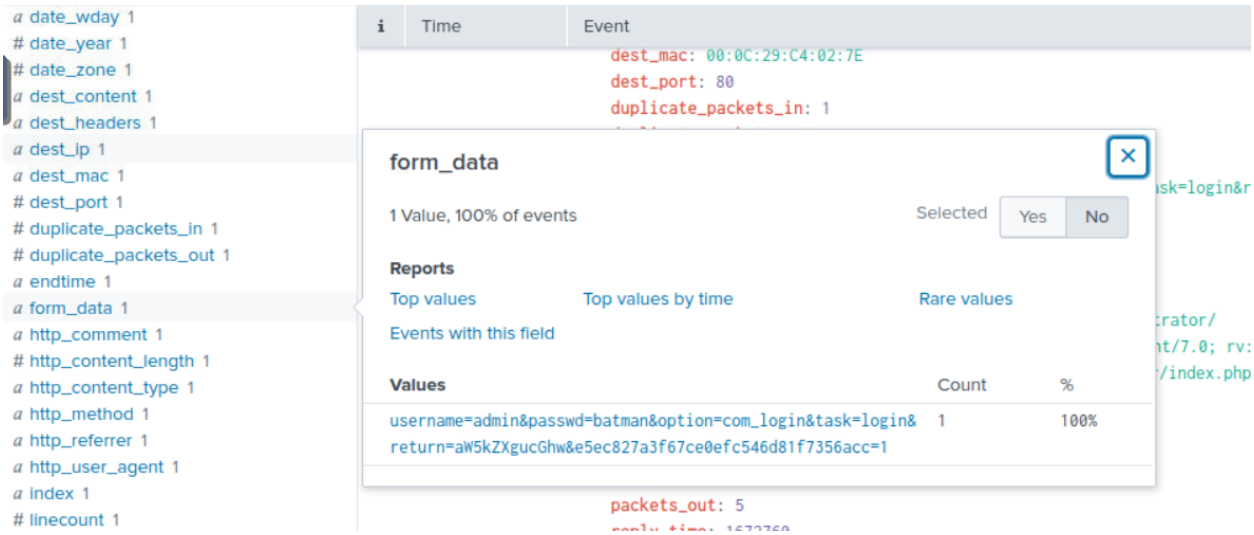


I used the query '**index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST form_data=*username*passwd* | rex field=form_data "passwd=(?<creds>\w+)" |table _time src_ip uri http_user_agent creds**', but this time appending **| table_time src_ip uri http_user_agent creds**. This query displayed 2 IP addresses responsible for the attempts. (23.22.63.114) which was used automated using Python-urllib/2.7 as the http_user_agent on our server as well as (40.80.148.42) which performed a single attempt using Mozilla/5.0 browser. The password used for this attempt was batman.



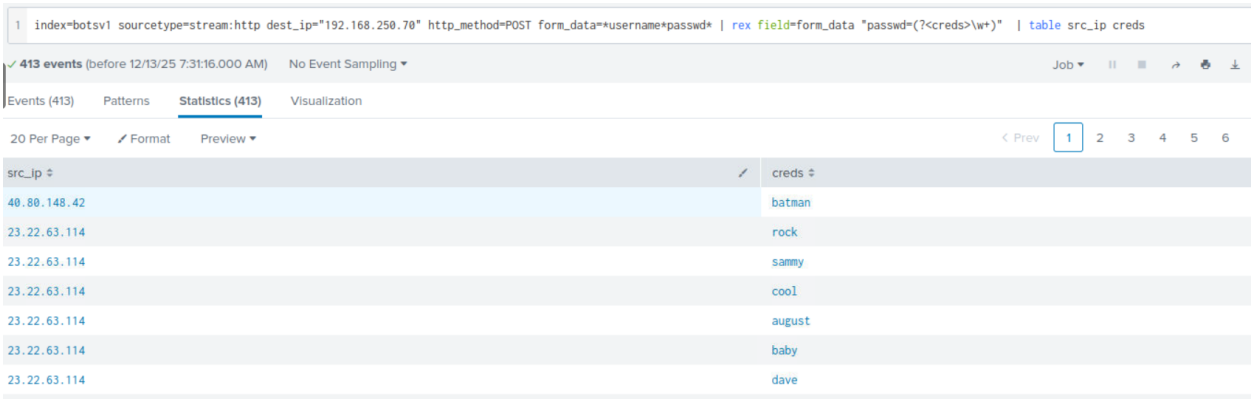
I clicked on batman under the filter '**creds**' and clicked on 'view events'. I went to the left side of the page and scrolled through the filters to **form_data** where I found the username - **admin**.

STEP 3: What was the correct password for admin access to the content management system running **imreallynotbatman.com**?



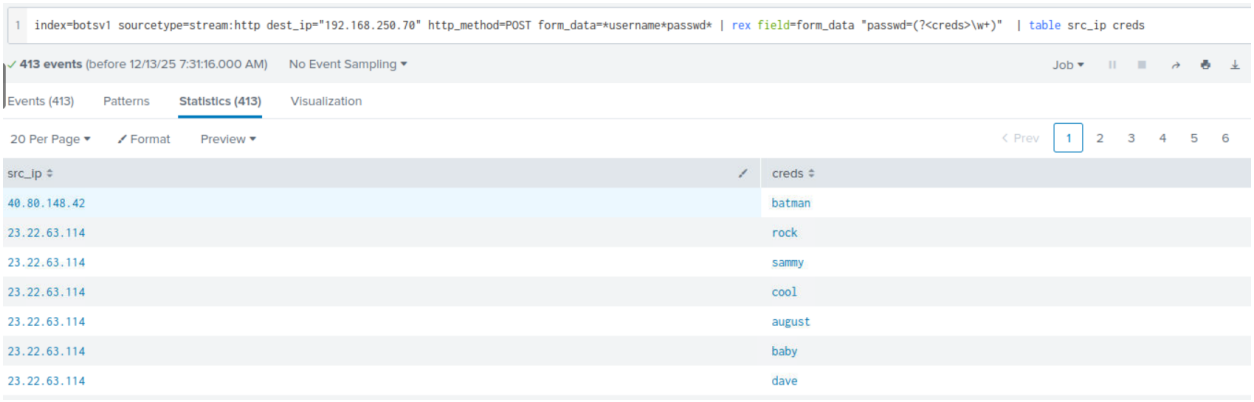
The correct password here was **batman**.

STEP 4: How many unique passwords were used to attempt the brute force attack?



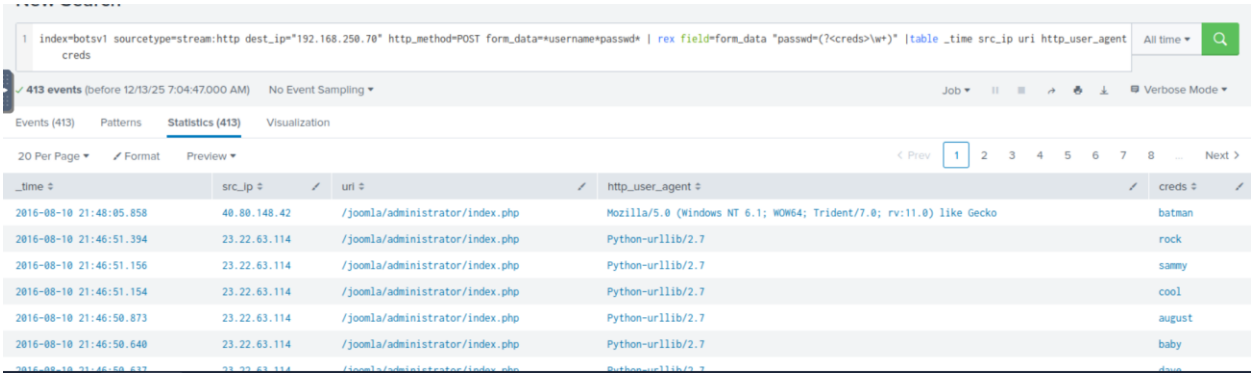
Looking at the stats. There were a total of '**412**' for the IP address 23.22.63.114.

STEP 6: What IP address is likely attempting a brute force password attack against **imreallynotbatman.com**?



The IP address is **'23.22.63.114'.**

STEP 7: After finding the correct password, which IP did the attacker use to log in to the admin panel?



Going back to our first step. I saw that the only IP that was able to do this was – **'40.80.148.42'**

Conclusion Summary

During the exploitation phase, Splunk was used to analyze HTTP POST traffic and identify a brute-force attack targeting the Joomla administrator login page. By examining form data and credential submission attempts, the investigation confirmed repeated password attempts against the admin account. One IP conducted large-scale automated brute-forcing, while a second IP successfully authenticated using the correct credentials. This analysis confirmed that the attacker gained administrative access to the CMS, completing the exploitation stage of the attack lifecycle.