

[Open in app ↗](#)**Medium**

Search



★ Get unlimited access to the best of Medium for less than \$1/week. [Become a member](#)



TryHackMe| Incident handling with Splunk



Mohamed Ashraf · Following

24 min read · May 2, 2023

[Listen](#)[Share](#)[More](#)

Task 1 Introduction: Incident Handling

This room covers an incident Handling scenario using Splunk. An incident from a security perspective is “Any event or action, that has a negative consequence on the security of a user/computer or an organization is considered a security incident.” Below are a few of the events that would negatively affect the environment when they occurred:

- Crashing the system
- Execution of an unwanted program
- Access to sensitive information from an unauthorized user
- A Website being defaced by the attacker

- The use of USB devices when there is a restriction in usage is against the company's policy



Learning Objective

- Learn how to leverage OSINT sites during an investigation
- How to map Attacker's activities to Cyber Kill Chain Phases
- How to utilize effective Splunk searches to investigate logs
- Understand the importance of host-centric and network-centric log sources

Room Prerequisites

Before going through this room, it is expected that the participants will have a basic understanding of Splunk. If not, consider going through this room, Splunk 101 (<https://tryhackme.com/jr/splunk101>).

It is good to understand the following before completing this lesson:

- Splunk overview and basic navigation
- Important Splunk Queries
- Know how to use different functions/values to craft a search query
- How to look for interesting fields

Task 2 Incident Handling — Life Cycle

Incident Handling Life Cycle

As an Incident Handler / SOC Analyst, we would aim to know the attackers' tactics, techniques, and procedures(TTPs). Then we can stop/defend/prevent against the attack in a better way. The Incident Handling process is divided into four different phases. Let's briefly go through each phase before jumping into the incident, which we will be going through in this exercise.

1. Preparation

The preparation phase covers the readiness of an organization against an attack. That means documenting the requirements, defining the policies, incorporating the

security controls to monitor like EDR / SIEM / IDS / IPS, etc. It also includes hiring/training the staff.

2. Detection and Analysis

The detection phase covers everything related to detecting an incident and the analysis process of the incident. This phase covers getting alerts from the security controls like SIEM/EDR investigating the alert to find the root cause. This phase also covers hunting for the unknown threat within the organization.

3. Containment, Eradication, and Recovery

This phase covers the actions needed to prevent the incident from spreading and securing the network. It involves steps taken to avoid an attack from spreading into the network, isolating the infected host, clearing the network from the infection traces, and gaining control back from the attack.

4. Post-Incident Activity / Lessons Learnt

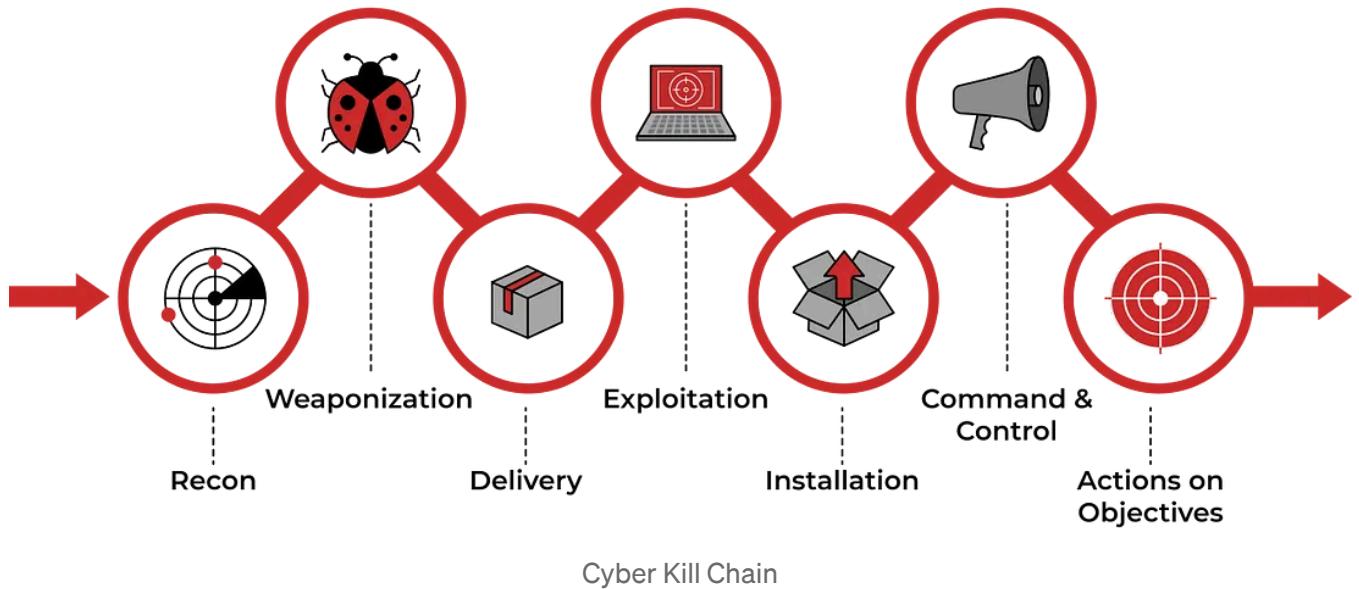
This phase includes identifying the loopholes in the organization's security posture, which led to an intrusion, and improving so that the attack does not happen next time. The steps involve identifying weaknesses that led to the attack, adding detection rules so that similar breach does not happen again, and most importantly, training the staff if required.

Task 3 Incident Handling: Scenario

In this exercise, we will investigate a cyber attack in which the attacker defaced an organization's website. This organization has Splunk as a SIEM solution setup. Our task as a Security Analyst would be to investigate this cyber attack and map the attacker's activities into all 7 of the Cyber Kill Chain Phases. It is important to note that we don't need to follow the sequence of the cyber kill chain during the Investigation. One finding in one phase will lead to another finding that may have mapped into some other phase.

Cyber Kill Chain

We will follow the Cyber kill Chain Model and map the attacker's activity in each phase during this Investigation. When required, we will also utilize Open Source Intelligence (OSINT) and other findings to fill the gaps in the kill chain. It is not necessary to follow this sequence of the phases while investigating.



- Reconnaissance
- Weaponization
- Delivery
- Exploitation
- Installation
- Command & Control
- Actions on Objectives

Scenario

A Big corporate organization Wayne Enterprises has recently faced a cyber-attack where the attackers broke into their network, found their way to their web server, and have successfully defaced their website <http://www.imreallynotbatman.com>. Their website is now showing the trademark of the attackers with the message **YOUR SITE HAS BEEN DEFACED** as shown below.

YOUR SITE HAS BEEN **DEFACED**

P01s0n1vy was HERE

Deal with it, Admin



trademark of the attackers with the message

They have requested “US” to join them as a **Security Analyst** and help them investigate this cyber attack and find the root cause and all the attackers’ activities within their network.

The good thing is, that they have Splunk already in place, so we have got all the event logs related to the attacker’s activities captured. We need to explore the records and find how the attack got into their network and what actions they performed.

This Investigation comes under the **Detection and Analysis** phase.

Splunk:

During our investigation, we will be using **Splunk** as our **SIEM** solution. Logs are being ingested from webserver/firewall/Suricata/Sysmon etc. In the data summary tab, we can explore the log sources showing visibility into both **network-centric** and **host-centric** activities. To get the complete picture of the hosts and log sources being monitored in Wayne Enterprise, please click on the **Data summary** and navigate the available tabs to get the information.

The screenshot shows the Splunk search interface. At the top, there are tabs for 'Search', 'Datasets', 'Reports', 'Alerts', and 'Dashboards'. On the right, there's a 'Search & Reporting' button. Below the tabs, there's a search bar with placeholder text 'enter search here...', a time range selector ('All time'), and a search icon. A note says 'No Event Sampling'. To the right, there's a summary of events: '956,379 Events INDEXED' (5 years ago, EARLIEST EVENT), 'Data Summary', and 'a few seconds ago LATEST EVENT'. Below the summary, there are links for 'Documentation' and 'Tutorial'. At the bottom left, there's a link to 'Search History'.

Interesting log Sources

Some of the interesting log sources that will help us in our Investigation are:

Log Sources	Details
wineventlog	It contains Windows Event logs
winRegistry	It contains the logs related to registry creation / modification / deletion etc.
XmlWinEventLog	It contains the sysmon event logs. It is a very important log source from an investigation point of view.
Fortigate_utm	It contains Fortinet Firewall logs
iis	It contains IIS web server logs
Nessus:scan	It contains the results from the Nessus vulnerability scanner.
Suricata	It contains the details of the alerts from the Suricata IDS. This log source shows which alert was triggered and what caused the alert to get triggered—a very important log source for the investigation.
stream:http	It contains the network flow related to http traffic.
stream:DNS	It contains the network flow related to DNS traffic.
stream:icmp	It contains the network flow related to icmp traffic.

Note: All the event logs that we are going to investigate are present in `index=botsv1`

Now that we know what hosts we have to investigate, what sources and the source types are, let's connect to the lab and start Investigating.

Task 4 : Reconnaissance Phase

Reconnaissance Phase



Reconnaissance is an attempt to discover and collect information about a target. It could be knowledge about the system in use, the web application, employees or location, etc.

We will start our analysis by examining any reconnaissance attempt against the webserver `imreallynotbatman.com`. From an analyst perspective, where do we first need to look? If we look at the available log sources, we will find some log sources covering the network traffic, which means all the inbound communication towards our web server will be logged into the log source that contains the web traffic. Let's start by searching for the domain in the search head and see which log source includes the traces of our domain.

Search Query: `index=botsv1 imreallynotbatman.com`

Search Query explanation: We are going to look for the event logs in the index “botsv1” which contains the term `imreallynotbatman.com`

The screenshot shows the Splunk Enterprise search interface. At the top, there is a navigation bar with links for 'splunk>enterprise' (highlighted), 'App: Search & Reporting', 'Messages', 'Settings', 'Activity', 'Help', and a search bar. Below the navigation bar is a search bar with placeholder text 'enter search here...' and a 'Last 24 hours' dropdown. To the right of the search bar are buttons for 'Smart Mode' and a magnifying glass icon. Underneath the search bar is a 'Search' section with a 'How to Search' link and a note about documentation. On the right side of the search bar, there is a 'What to Search' summary table:

957,020 Events INDEXED	5 years ago EARLIEST EVENT	a few seconds ago LATEST EVENT
Data Summary		

At the bottom left, there is a 'Search History' link.

Here we have searched for the term `imreallynotbatman.com` in the index `botsv1`. In the sourcetype field, we saw that the following log sources contain the traces of this search term.

- Suricata
- stream:http
- fortigate_utm
- iis

From the name of these log sources, it is clear what each log source may contain. Every analyst may have a different approach to investigating a scenario. Our first task is to identify the IP address attempting to perform reconnaissance activity on our web server. It would be obvious to look at the web traffic coming into the network. We can start looking into any of the logs mentioned above sources.

Let us begin looking at the log source `stream:http`, which contains the http traffic logs, and examine the `src_ip` field from the left panel. `Src_ip` field contains the source IP address it finds in the logs.

Search Query: `index=botsv1 imreallynotbatman.com sourcetype=stream:http`

Search Query Explanation: This query will only look for the term `imreallynotbatman.com` in the `stream:http` log source.

1 index=botsv1 imreallynotbatman.com sourcetype=stream:http

✓ 22,200 events (before 1/21/22 6:21:51.000 AM) No Event Sampling ▾

Events (22,200) Patterns Statistics Visualization

Format Timeline ▾ - Zoom Out + Zoom to Selection × Deselect

List ▾ Format 20 Per Page ▾

< Hide Fields : All Fields

SELECTED FIELDS

- host 1
- source 1
- sourcetype 1
- src_ip 2**

INTERESTING FIELDS

- accept 9
- ack_packets_in 41
- ack_packets_out 12
- action 2
- app 1

src_ip

2 Values, 99.856% of events Selected Yes No

Reports

Top values Top values by time Rare values

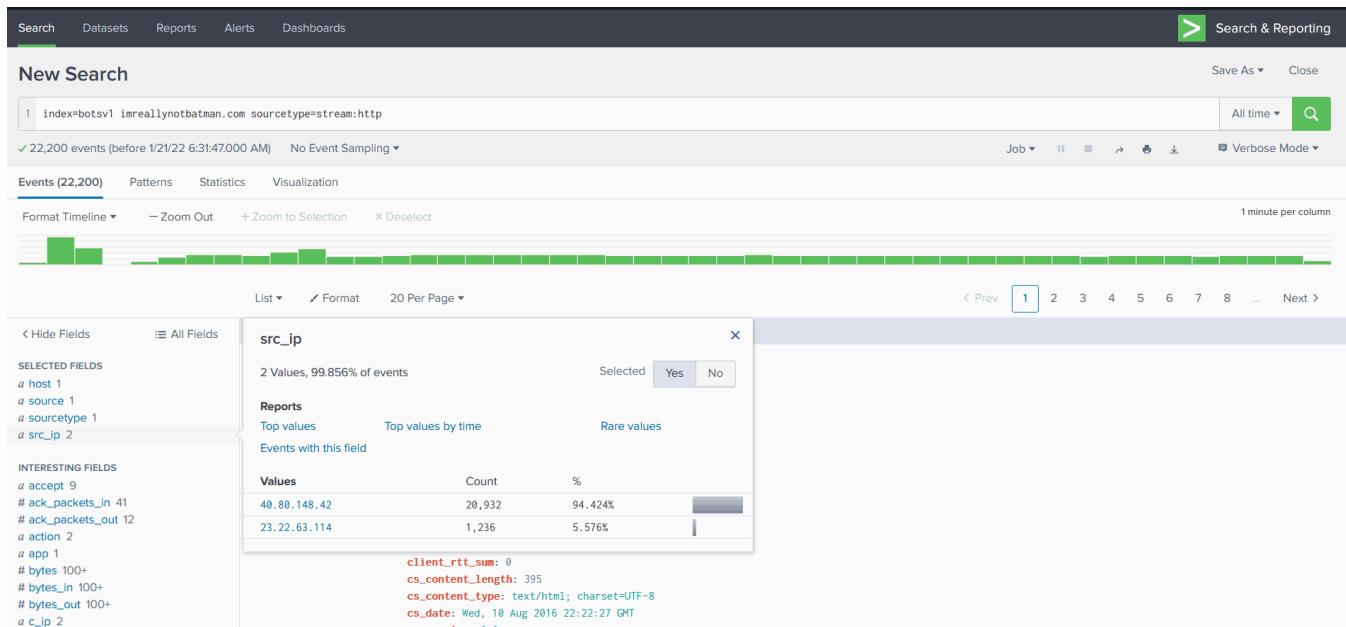
Events with this field

Values	Count	%
40.80.148.42	20,932	94.424%
23.22.63.114	1,236	5.576%

Note: The important thing to note, if you don't find the field of interest, keep scrolling in the left panel. When you click on a field, it will contain all the values it finds in the logs.

So far, we have found two IPs in the src_ip field 40.80.148.42 and 23.22.63.114 . The first IP seems to contain a high percentage of the logs as compared to the other IP, which could be the answer. If you want to confirm further, click on each IP one by one, it will be added into the search query, and look at the logs, and you will find the answer.

To further confirm our suspicion about the IP address 40.80.148.42, click on the IP and examine the logs. We can look at the interesting fields like User-Agent, Post request, URIs, etc., to see what kind of traffic is coming from this particular IP.



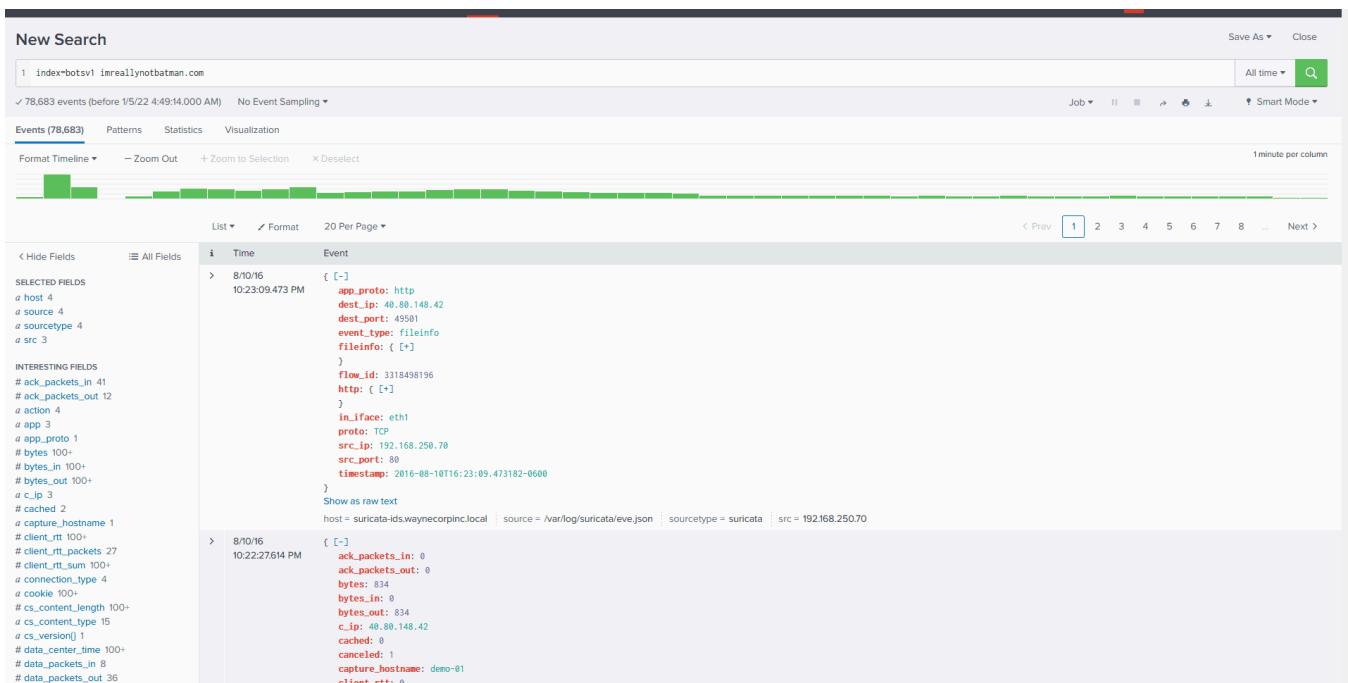
We have narrowed down the results to only show the logs from the source IP **40.80.148.42**, looked at the fields of interest and found the traces of the domain being probed.

Validate the IP that is scanning

So what do we need to do to validate the scanning attempt? Simple, dig further into the weblogs. Let us narrow down the result, look into the `suricata` logs, and see if any rule is triggered on this communication.

Search Query: `index=botsv1 imreallynotbatman.com src=40.80.148.42 sourcetype=suricata`

Search Query Explanation: This query will show the logs from the suricata log source that are detected/generated from the source IP **40.80.148.42**



We have narrowed our search on the `src IP` and looked at the source type `suricata` to see what Suricata triggered alerts. In the right panel, we could not find the field of our interest, so we clicked on more fields and searched for the fields that contained the signature alerts information, which is an important point to note.

Answer the questions below :

1-One suricata alert highlighted the CVE value associated with the attack attempt.

What is the CVE value?

alert.signature

46 Values, 100% of events

Selected

Yes

No

Reports[Top values](#)[Top values by time](#)[Rare values](#)[Events with this field](#)**Top 10 Values**

Count

%

ET WEB_SERVER Script tag in URI, Possible Cross Site Scripting Attempt	103	21.776%
ET WEB_SERVER Onmouseover= in URI - Likely Cross Site Scripting Attempt	48	10.148%
ET WEB_SERVER Possible XXE SYSTEM ENTITY in POST BODY.	41	8.668%
SURICATA HTTP Host header invalid	35	7.4%
ET WEB_SERVER Possible SQL Injection Attempt SELECT FROM	33	6.977%
ET WEB_SERVER SQL Injection Select Sleep Time Delay	32	6.765%
ET WEB_SERVER Possible CVE-2014-6271 Attempt	18	3.805%
ET WEB_SERVER Possible CVE-2014-6271 Attempt in Headers	18	3.805%
ET WEB_SERVER PHP tags in HTTP POST	13	2.748%

Answer: CVE-2014-6271**2-What is the CMS our web server is using?**Found in the http.http_refer and http.url fields a site called **joomla**

<input type="checkbox"/> http.http_refer ▾	http://imreallynotbatman.com/joomla/administrator/index.php?option=com_explorer&tmpl=component	▼
<input type="checkbox"/> http.http_user_agent ▾	Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko	▼
<input type="checkbox"/> http.length ▾	0	▼
<input type="checkbox"/> http.protocol ▾	HTTP/1.1	▼
<input type="checkbox"/> http.url ▾	/joomla/administrator/index.php	▼
<input type="checkbox"/> http_method ▾	POST	▼
<input type="checkbox"/> http_protocol ▾	HTTP/1.1	▼
<input type="checkbox"/> http_referrer ▾	http://imreallynotbatman.com/joomla/administrator/index.php?option=com_explorer&tmpl=component	▼

After researching what this is, it turned out to be “CMS”

The Joomla! website homepage. The main banner features a group of diverse people smiling and waving, with the text "The Flexible Platform Empowering Website Creators". Below the banner, a subtext states: "Joomla! is an award-winning content management system (CMS), which enables you to build web sites and powerful online applications."

Answer: joomla

3-What is the web scanner, the attacker used to perform the scanning attempts?

Most of the scanning tools are logged under “user-agent”

```
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.21 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.21
Acunetix-Product: WVS/10.0 [Acunetix] Web Vulnerability Scanner - Free Edition)
Acunetix-Scanning-agreement: Third Party Scanning PROHIBITED
Acunetix-User-agreement: http://www.acunetix.com/wvs/disc.htm
Accept: */*
```

Answer: Acunetix

4-What is the IP address of the server imreallynotbatman.com?

Event	<input type="checkbox"/> bytes ▾	16684
	<input type="checkbox"/> dest ▾	imreallynotbatman.com
	<input type="checkbox"/> dest_ip ▾	192.168.250.70

Answer: 192.168.250.70

Task 5: Exploitation Phase

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.

Count

Let's use the following search query to see the number of counts by each source IP against the webserver.

Search Query: `index=botsv1 imreallynotbatman.com sourcetype=stream* | stats count(src_ip) as Requests by src_ip | sort - Requests`

Query Explanation: This query uses the stats function to display the count of the IP addresses in the field `src_ip`.

src_ip	Requests
40.80.148.42	17483
23.22.63.114	1235

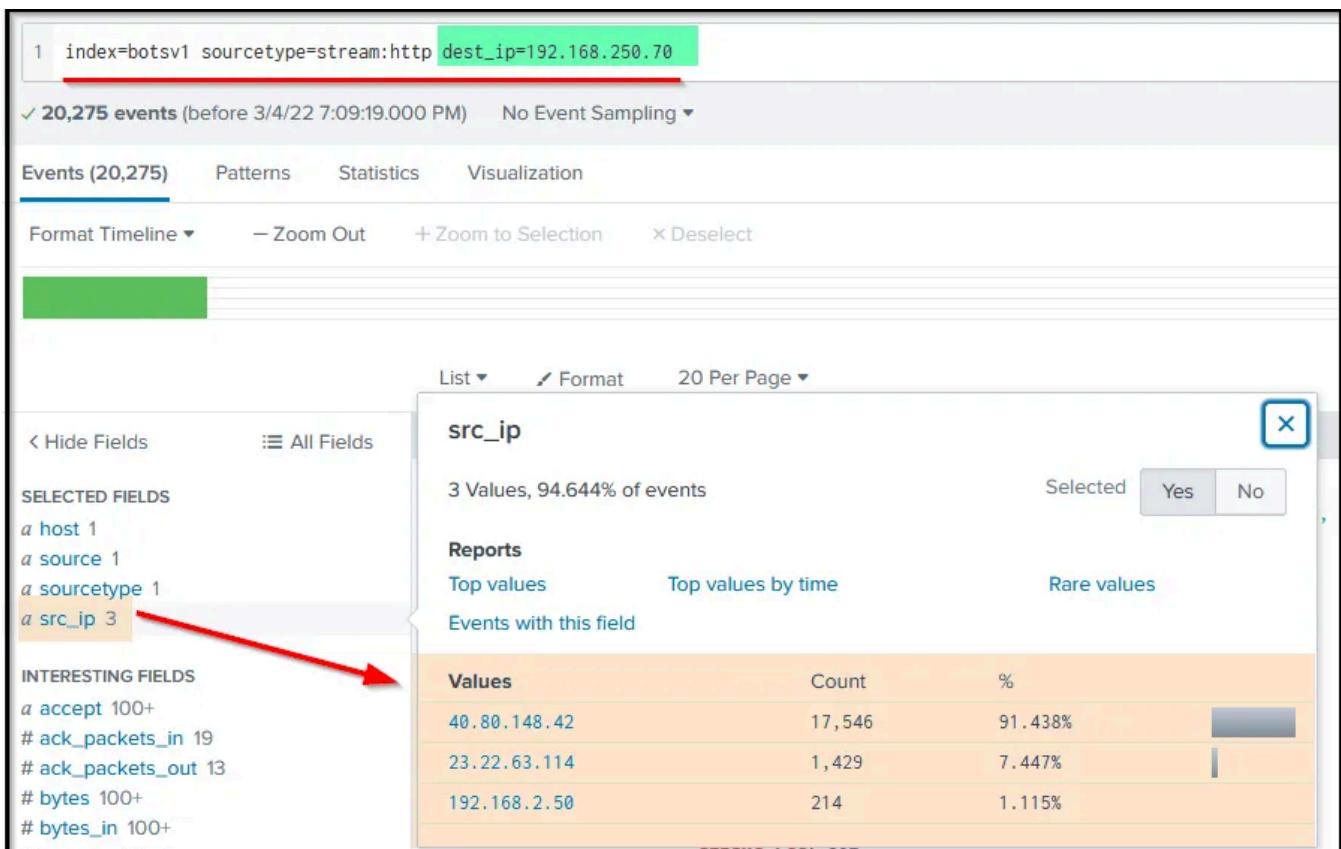
Additionally, we can also create different visualization to show the result. Click on **Visualization → Select Visualization** as shown below.



Now we will narrow down the result to show requests sent to our web server, which has the IP 192.168.250.70

Search Query: index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70"

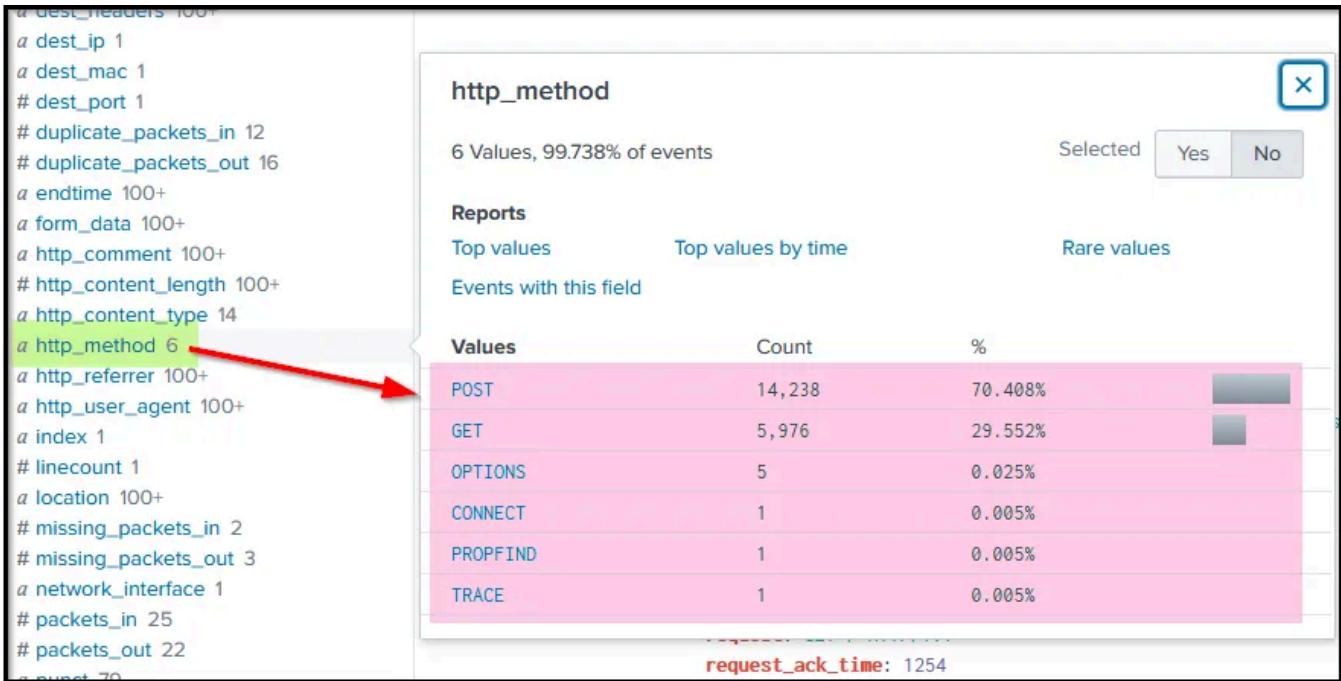
Query Explanation: This query will look for all the inbound traffic towards IP 192.168.250.70.



The result in the src_ip field shows three IP addresses (1 local IP and two remote IPs) that originated the HTTP traffic towards our webserver.

Another interesting field, `http_method` will give us information about the HTTP Methods observed during these HTTP communications.

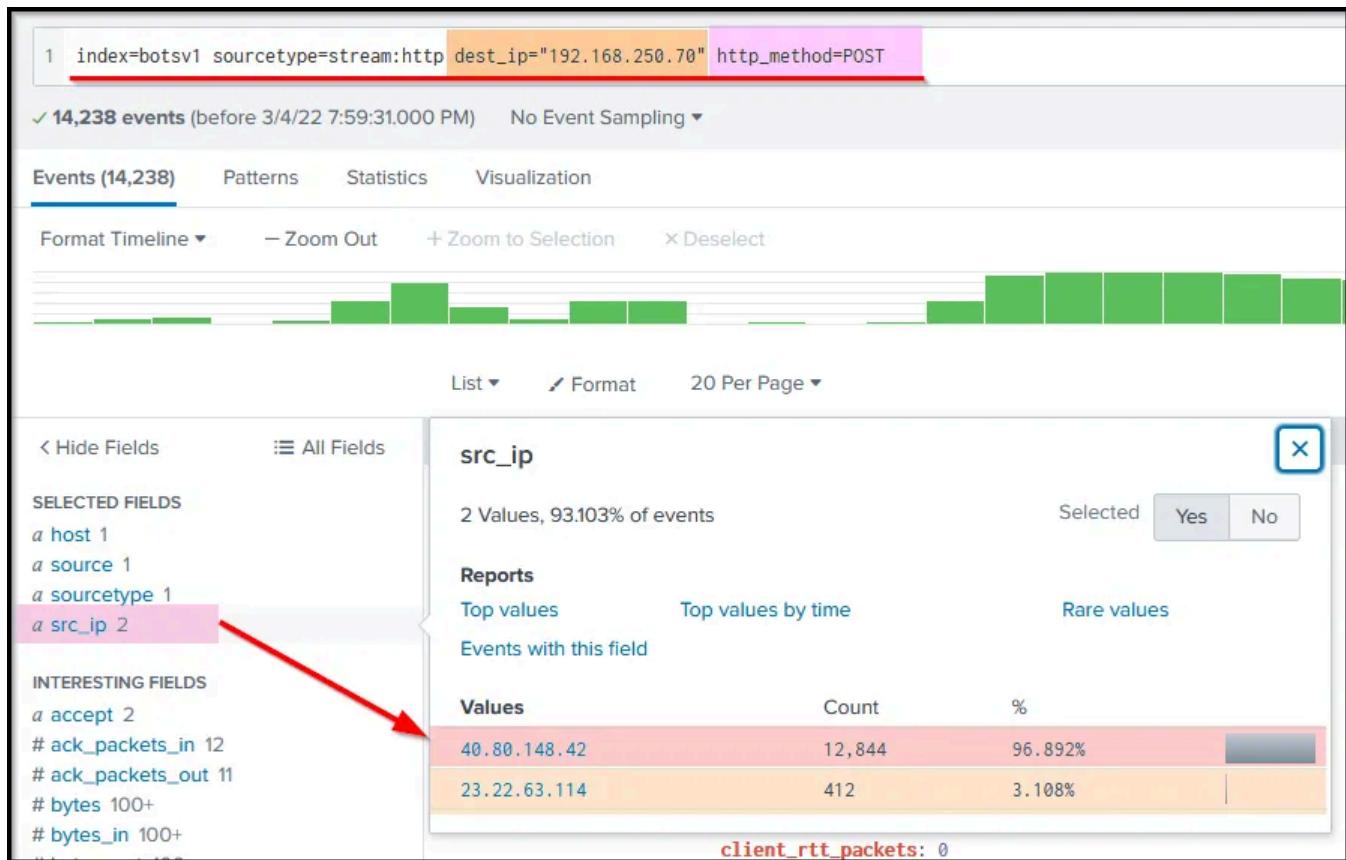
We observed most of the requests coming to our server through the POST request, as shown below.



To see what kind of traffic is coming through the POST requests, we will narrow down on the field `http_method=POST` as shown below:

Search Query: `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70"`

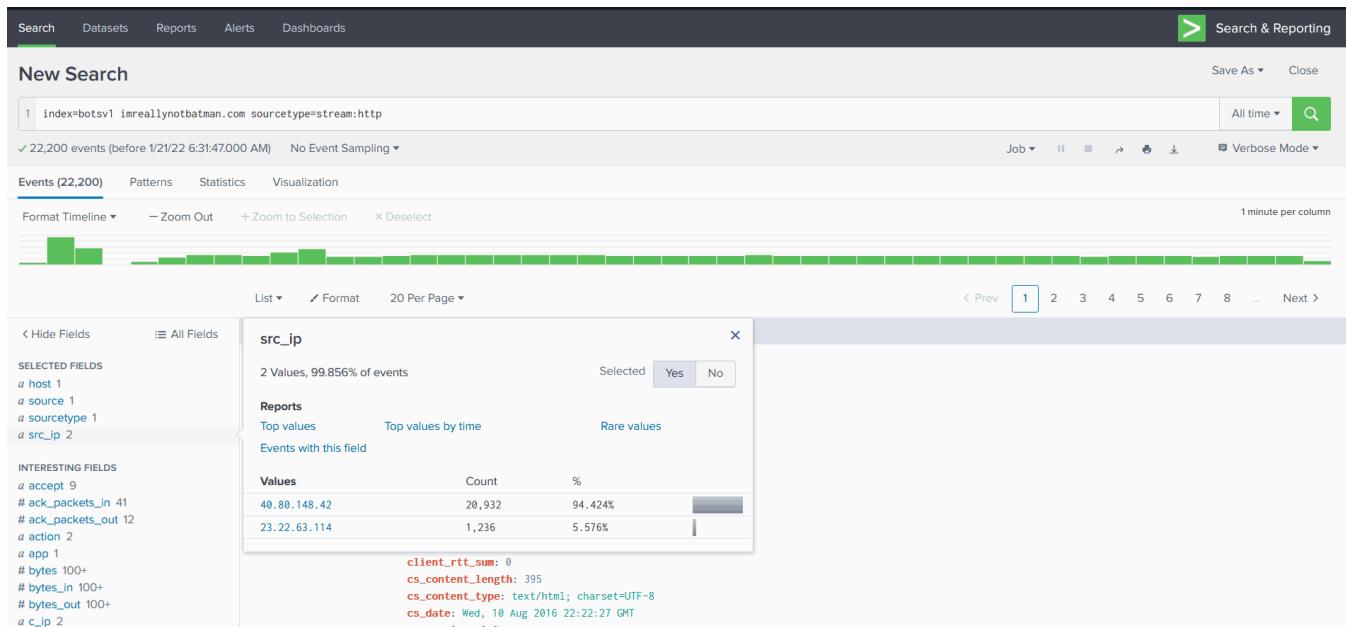
`http_method=POST`



The result in the `src_ip` field shows two IP addresses sending all the POST requests to our server.

Interesting fields: In the left panel, we can find some interesting fields containing valuable information. Some of the fields are:

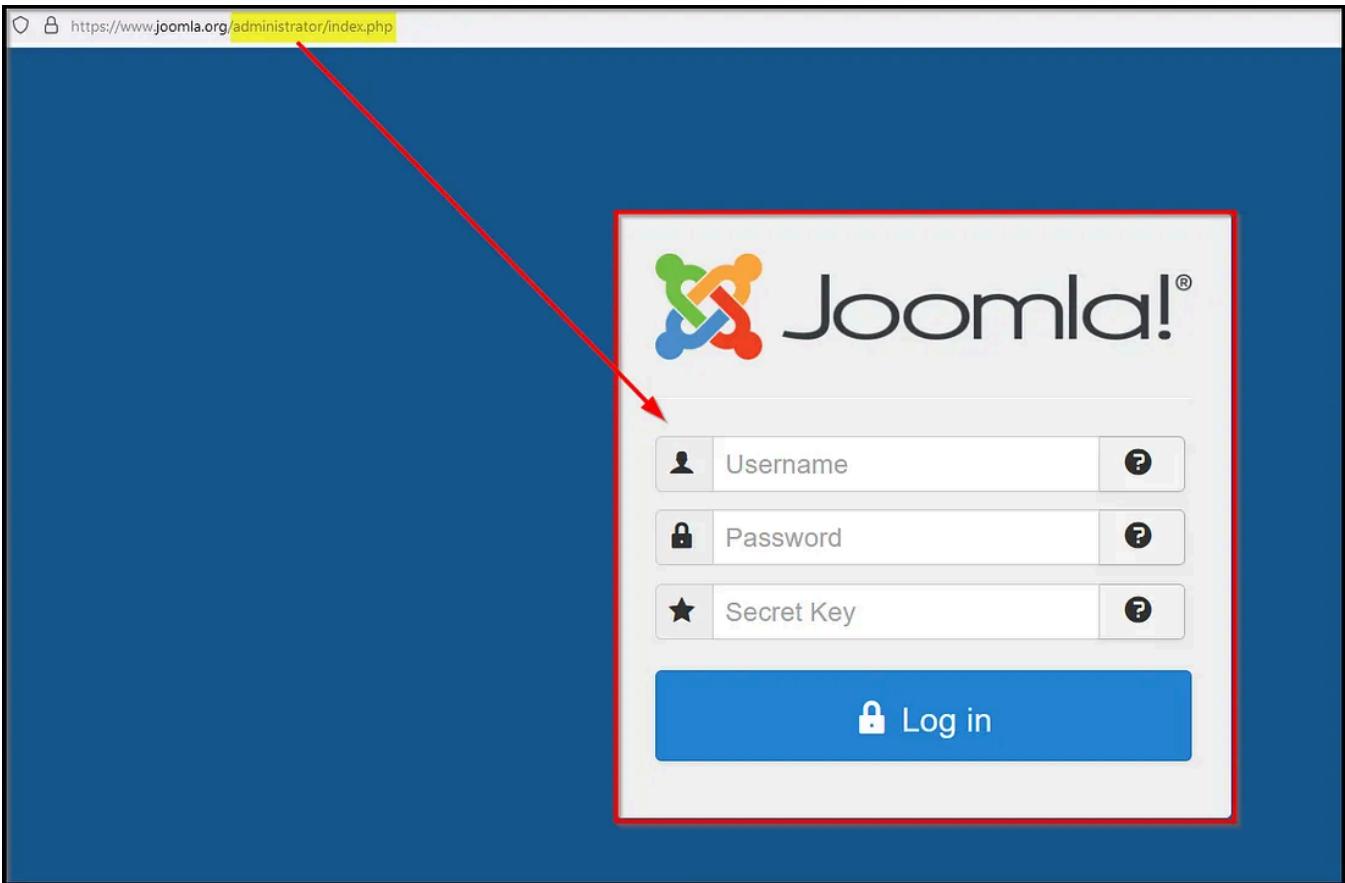
- `src_ip`
- `form_data`
- `http_user_agent`
- `uri`



The term Joomla is associated with the webserver found in a couple of fields like **uri**, **uri_path**, **http_referrer**, etc. This means our webserver is using Joomla CMS (Content Management Service) in the backend.

A little search on the internet for the admin login page of the Joomla CMS will show as -> /joomla/administrator/index.php

It is important because this uri contains the login page to access the web portal therefore we will be examining the traffic coming into this admin panel for a potential brute-force attack.



Reference:

<https://www.joomla.org/administrator/index.php>

We can narrow down our search to see the requests sent to the login portal using this information.

Search query: index=botsv1 imreallynotbatman.com sourcetype=stream:http
dest_ip="192.168.250.70" uri="/joomla/administrator/index.php"

Query Explanation: We are going to add `uri="/joomla/administrator/index.php"` in the search query to show the traffic coming into this URI.

The screenshot shows the Splunk interface with a search result for the 'form_data' field. The search query at the top is: `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" uri="/joomla/administrator/index.php"`. Below the search bar, it says '1,253 events (before 1/21/22 8:12:53.000 AM) No Event Sampling ▾'. The main area shows a timeline from 8/10/16 to 8/10/16. The results table has columns: Time, Event, and a sidebar for 'Selected' (Yes or No). The 'Selected' column for 'form_data' is set to 'Yes'. The table shows the top 10 values for 'form_data':

Value	Count	%
start=0&limit=150&dir=/joomla&option=com_extplorer&action=getdircontents&sendWhat=both	4	0.934%
action=chdir_event&dir=&option=com_extplorer	2	0.467%
action=chdir_event&dir=/joomla&option=com_extplorer	1	0.234%
action=chdir_event&dir=ext_root&option=com_extplorer	1	0.234%

form_data The field contains the requests sent through the form on the admin panel page, which has a login page. We suspect the attacker may have tried multiple credentials in an attempt to gain access to the admin panel. To confirm, we will dig deep into the values contained within the `form_data` field, as shown below:

Search Query: `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70"`

```
http_method=POST uri="/joomla/administrator/index.php" | table _time uri src_ip
dest_ip form_data
```

Query Explanation: We will add this `-> | table _time uri src_ip dest_ip form_data` to create a table containing important fields as shown below:

index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST uri="/joomla/administrator/index.php" table _time uri src_ip dest_ip form_data						Job	Smart Mode				
425 events (before 3/4/22 9:57:44:000 PM) No Event Sampling						1	2	3	4	5	Next
Events Patterns Statistics (425) Visualization											
100 Per Page	Format	Preview									
_time *	uri *	src_ip *	dest_ip *	form_data *							
2016-08-10 21:45:21.226	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=123456789d873c2becd118318849d13cf18b60ff=1							
2016-08-10 21:45:21.241	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&n863349a657c211fbfe90ebe9427654c=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=letmein							
2016-08-10 21:45:21.247	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=qwerty&af4df60674155567dee0566f87045251=1							
2016-08-10 21:45:21.250	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=1234&aa6297ae5c1e3df78a421bc55548d16=1							
2016-08-10 21:45:21.260	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&76e93e848809a46878468d88954a0d54=1&passwd=123456							
2016-08-10 21:45:21.263	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=football&dd1181413b1a70460b8d425cec799cdca=1							
2016-08-10 21:45:21.325	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=pussy&b54cc9395cafef6db9cad73ceacde7=1							
2016-08-10 21:45:21.826	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=michael&bd5d773b68cd015b02121ff36a2c4a=1							
2016-08-10 21:45:22.000	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=master&aa5d789b1af633e3de4b3dc95daa1877=1							
2016-08-10 21:45:22.002	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=superman&cab483edaec06d4e888547d3f57becb=1							
2016-08-10 21:45:22.005	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&c77289d571cfe66a59479cb2706f4a=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=1234567							
2016-08-10 21:45:22.008	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=shadow&5dd67f80e801cd4f24920dae0fb2fd=1							
2016-08-10 21:45:22.012	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=dragon&45b663f3374634ca3fd8601177601c=1							
2016-08-10 21:45:22.034	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&ad30d20bb0b8474d4a81199d95be395f=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=111111							
2016-08-10 21:45:22.063	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&a462a0006d3b8338ba4d11967ceec0=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=baseball							
2016-08-10 21:45:22.570	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&d9df6e24cc184413a74e1025ad274cd2=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=mustang							
2016-08-10 21:45:22.710	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=jennifer&adc1c15a0951d0ff17e2f4cafa4f292ff9=1							

If we keep looking at the results, we will find two interesting fields `username` that includes the single username `admin` in all the events and another field `passwd` that contains multiple passwords in it, which shows the attacker from the IP `23.22.63.114` Was trying to guess the password by brute-forcing and attempting numerous passwords.

The time elapsed between multiple events also suggests that the attacker was using an automated tool as various attempts were observed in a short time.

Extracting Username and Passwd Fields using Regex

Looking into the logs, we see that these fields are not parsed properly. Let us use Regex in the search to extract only these two fields and their values from the logs and display them.

We can display only the logs that contain the `username` and `passwd` values in the `form_data` field by adding `form_data=*username*passwd*` in the above search.

Search Query: `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70"`

```
http_method=POST uri="/joomla/administrator/index.php" form_data=*username*passwd*
| table _time uri src_ip dest_ip form_data
```

1 index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST uri="/joomla/administrator/index.php" form_data=*username*passwd* | table _time uri src_ip dest_ip form_data

✓ 413 events (before 3/4/22 10:17:56.000 PM) No Event Sampling ▾

Events Patterns Statistics (413) Visualization

100 Per Page ▾ Format Preview ▾

_time	uri	src_ip	dest_ip	form_data
2016-08-10 21:45:21.325	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=pussy&b4cc9395cafcef6dab9cad73ceacd7=1
2016-08-10 21:46:05.858	/Joomla/administrator/index.php	40.80.148.42	192.168.250.70	username=admin&passwd=batman&option=com_login&task=login&return=aW5kZXgucGhw&e5ec827a3f67ce0efc546d81f7356acc=1
2016-08-10 21:46:51.394	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=rock&a40c518220c1993f0e02dc4712c5794=1
2016-08-10 21:46:51.154	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=cool&a9349d0d6bd0f078ad72cf8e9348583=1
2016-08-10 21:46:51.156	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=sammy&dd3bb0020f7004affba32f7d0fa7fa88=1
2016-08-10 21:46:50.873	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=august&9800c58b682f234e562dee5972a58b8d=1
2016-08-10 21:46:50.634	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=phantom&a083bf4d12c07976186d8a6efa6308cf=1
2016-08-10 21:46:50.627	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=williams&e3b1998d29669e83333a01735fdic90=1
2016-08-10 21:46:50.621	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&ba1501d963f628dfb862d3a07bbe674+&task=login&return=aW5kZXgucGhw&option=com_login&passwd=private
2016-08-10 21:46:50.640	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=baby&26a9247d113c378cdf06f31fa2154f2c=1
2016-08-10 21:46:50.637	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=dave&1b067a8762b4c8a9909ca68aae723e5a=1
2016-08-10 21:46:50.632	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&e6ca600ce3bd81316681686dfbd0a5=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=donald
2016-08-10 21:46:50.629	/Joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=lifehack&5804a636e6c85901f132655dae4add9b=1

It's time to use **Regex (regular expressions)** to extract all the password values found against the field passwd in the logs. To do so, Splunk has a function called rex. If we type it in the search head, it will show detail and an example of how to use it to extract the values.

2 | rex

✓ 413

Event rex field=GuestDNSName...ost>[^.]*\.[a-zA-Z]" Command History

rex field=Local ".:(?<port>.*)" Command History

rex field=_raw "total...s+'(?<freemem>[^']*'" Command History

Event rex field=_raw "search...[?(?<search>.*)] \। " Command History

rex field=action "(?<c...ge>add|delete|update)" Command History

100 F

rex

Specifies a Perl regular expression named groups to extract fields while you search.

Learn More ↗

Example:

... | rex field=_raw "From: (?<from>.* To: (?<to>.*"

Now, let's use Regex. **rex field=form_data "passwd=(?<creds>\w+)"** To extract the passwd values only. This will pick the form_data field and extract all the values found with the field. **creds**.

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=(?\w+)" | table src_ip creds
```

The screenshot shows a Splunk search interface with the following details:

- Search Bar:** index=botsv1 sourcetype=http form_data=*username*passwd* | table _time form_data
- Job Status:** 413 events (before 1/21/22 8:46:43:00 AM) No Event Sampling
- Event Count:** Events (413)
- Statistics:** Statistics (413) (selected)
- Visualizations:** Visualization
- Page Options:** 100 Per Page, Format, Preview, Job, II, III, IV, V, VI, VII, VIII, IX, All time, Verbose Mode
- Result Preview:** Shows log entries starting with the first few lines.

We have extracted the passwords being used against the username admin on the admin panel of the webserver. If we examine the fields in the logs, we will find two values against the field `http_user_agent` as shown below:

The screenshot shows a Splunk search results page for the `http_user_agent` field. The results are as follows:

Value	Count	%
Python-urllib/2.7	412	99.758%
Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko	1	0.242%

The first value clearly shows attacker used a python script to automate the brute force attack against our server. But one request came from a Mozilla browser. WHY? To find the answer to this query, let's slightly change to the about search query and add `http_user_agent` a field in the search head.

Let's create a table to display key fields and values by appending `-> | table _time src_ip uri http_user_agent creds` in the search query as shown below.

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=(?\w+)" |table _time src_ip uri http_user_agent creds
```

1 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</creds>				All time ▾
✓ 413 events (before 3/4/22 11:41:36.000 PM) No Event Sampling ▾				Job ▾
Events	Patterns	Statistics (413)	Visualization	Smart Mode ▾
100 Per Page ▾	Format	Preview ▾		< Prev 1 2 3 4 5 Next >
_time ▾	src_ip ▾	http_user_agent ▾		✓ creds ▾
2016-08-10 21:48:05.858	40.80.148.42	Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko		batman
2016-08-10 21:46:51.394	23.22.63.114	Python-urllib/2.7		rock
2016-08-10 21:46:51.156	23.22.63.114	Python-urllib/2.7		sammy
2016-08-10 21:46:51.154	23.22.63.114	Python-urllib/2.7		cool
2016-08-10 21:46:50.873	23.22.63.114	Python-urllib/2.7		august
2016-08-10 21:46:50.640	23.22.63.114	Python-urllib/2.7		baby
2016-08-10 21:46:50.637	23.22.63.114	Python-urllib/2.7		dave
2016-08-10 21:46:50.634	23.22.63.114	Python-urllib/2.7		phantom
2016-08-10 21:46:50.632	23.22.63.114	Python-urllib/2.7		donald
2016-08-10 21:46:50.629	23.22.63.114	Python-urllib/2.7		lifehack
2016-08-10 21:46:50.627	23.22.63.114	Python-urllib/2.7		williams
2016-08-10 21:46:50.624	23.22.63.114	Python-urllib/2.7		godzilla
2016-08-10 21:46:50.621	23.22.63.114	Python-urllib/2.7		private
2016-08-10 21:46:48.649	23.22.63.114	Python-urllib/2.7		4444
2016-08-10 21:46:47.775	23.22.63.114	Python-urllib/2.7		arthur

This result clearly shows a continuous brute-force attack attempt from an IP **23.22.63.114** and 1 password attempt **batman** from IP **40.80.148.42** using the Mozilla browser.

Answer the questions below :

1-What IP address is likely attempting a brute force password attack against **imreallynotbatman.com?**

40.80.148.42	/joomla/administrator/index.php	Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko	batman
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	rock
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	sammy
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	cool
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	august
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	baby
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	dave
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	phantom
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	donald
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	lifehack
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	williams
23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7	godzilla

Answer: 23.22.63.114

2-What was the URI which got multiple brute force attempts?

23.22.63.114	/joomla/administrator/index.php	Python-urllib/2.7

Answer: /joomla/administrator/index.php

3-Against which username was the brute force attempt made?

New Search						Save As ▾	Create Tab		
1 index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST uri="/joomla/administrator/index.php" form_data+=username+passwd+ table _time uri src_ip dest_ip form_data						Job ▾	II	☰	✖
✓ 413 events (before 3/4/22 10:17:56.000 PM) No Event Sampling ▾									
Events	Patterns	Statistics (413)	Visualization						
100 Per Page ▾	Format	Preview ▾					< Prev	1	2
_time	uri	src_ip	dest_ip	form_data					
2016-08-10 21:45:21.325	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=pussy&b4cc9395cafcef6dab9cad73ceacd7=1					
2016-08-10 21:48:05.858	/joomla/administrator/index.php	40.80.148.42	192.168.250.70	username=admin&passwd=batman&option=com_login&task=login&return=aW5kZXgucGhw&e5ec827a3f67ce0efc546d81f7356acc=1					
2016-08-10 21:46:51.394	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=rock&a40c518220c1993f0e02dc4712c5794=1					
2016-08-10 21:46:51.154	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=cool&a09349d0d6bdbf078ad72cf8e9348583=1					
2016-08-10 21:46:51.156	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=sammy&d3bb002f70044ffba32f7d0fa7fa88=1					
2016-08-10 21:46:50.873	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=august&9800c58b682f234e562dee5972a58b8d=1					
2016-08-10 21:46:50.634	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=phantom&a083bf4d12c07976186d8a6ef6a6308cf=1					
2016-08-10 21:46:50.627	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=wiliiams&e3b1998d29669e83333a101735fd1c90=1					
2016-08-10 21:46:50.621	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&bal11501d963f628dfb862d3a07bbe674=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=orivate					
2016-08-10 21:46:50.640	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=baby&26a9247d113c378cdf06f3fa2154f2c=1					
2016-08-10 21:46:50.637	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=dave&b067a8762b4ca9909ca68aae723e5a=1					
2016-08-10 21:46:50.632	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&beeca7600ce3bd81316681686dfbd0a5=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=donald					
2016-08-10 21:46:50.629	/joomla/administrator/index.php	23.22.63.114	192.168.250.70	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=lifehack&5804a636ec85901f132655dae4add9b=1					

Answer: Admin

4-What was the correct password for admin access to the content management system running imreallynotbatman.com?

when looking at web traffic if an attack was successful is Content-Length which is under src_headers in Splunk.

```
index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST f
```

Splunk > enterprise Apps ▾

Search Analytics Datasets Reports Alerts Dashboards

New Search

1 index=botsvl sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST form_data="username=passw&src_ip uri http_user_agent creds src_headers" | rex field=form_data "passwd=(?<creds>\w+)" | table _time

413 events (before 5/1/23 3:24:16.000 PM) No Event Sampling Job ▾ II III A B C D E F G H I J K L M N O P Q R S T V Verbose Mode ▾

Events (413) Patterns Statistics (413) Visualization

20 Per Page ▾ Format Preview ▾

_time	src_ip	uri	http_user_agent	creds	src_headers
2016-08-10 21:48:05.858	40.80.148.42	/joomla/administrator/index.php	Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko	batman	POST /joomla/administrator/index.php HTTP/1.1 Accept: text/html, application/xhtml+xml, */* Referer: http://imreallynotbatman.com/joomla/administrator/ Accept-Language: en-US Content-Type: application/x-www-form-urlencoded User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko Accept-Encoding: gzip, deflate Host: imreallynotbatman.com

Answer: batman

5-How many unique passwords were attempted in the brute force attempt?

```
index=botsvl sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST f
```

count() : a sql function which count how many things are returned and we want count how many password are returned.

As : is what we want to label the header.

New Search

1 index=botsvl sourcetype=stream:http dest_ip="192.168.250.70" http_method=POST form_data="username=passw&src_ip="23.22.63.114" | rex field=form_data "passwd=(?<creds>\w+)"

412 events (before 5/1/23 3:29:58.000 PM) No Event Sampling Job ▾ II III A B C D E F G H I J K L M N O P Q R S T V Verbose Mode ▾

Events (412) Patterns Statistics Visualization

Format Timeline ▾ - Zoom Out + Zoom to Selection X Deselect 1 second per column

List ▾ Format 20 Per Page ▾

Answer: 412

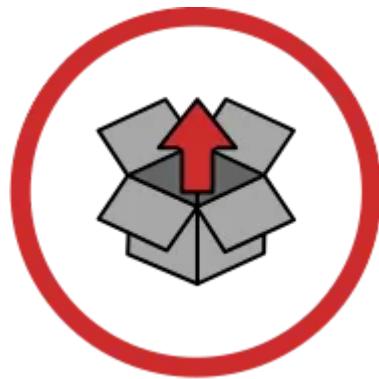
6-After finding the correct password, which IP did the attacker use to log in to the admin panel?

_time	src_ip	uri	http_user_agent	creds	src_headers
2016-08-10 21:48:05.858	40.80.148.42	/joomla/administrator/index.php	Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko	batman	POST /joomla/administrator/index.php HTTP/1.1 Accept: text/html, application/xhtml+xml, */* Referer: http://imreallynotbatman.com/joomla/administrator/ Accept-Language: en-US Content-Type: application/x-www-form-urlencoded User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko Accept-Encoding: gzip, deflate Host: imreallynotbatman.com Content-Length: 111

Answer: 40.80.148.42

Task 6: Installation Phase

Installation Phase



Once the attacker has successfully exploited the security of a system, he will try to install a backdoor or an application for persistence or to gain more control of the system. This activity comes under the installation phase.

In the previous Exploitation phase, we found evidence of the webserver `iamreallynotbatman.com` getting compromised via brute-force attack by the attacker using the python script to automate getting the correct password. The attacker used the IP" for the attack and the IP to log in to the server. This phase will investigate any payload / malicious program uploaded to the server from any attacker's IPs and installed into the compromised server.

To begin an investigation, we first would narrow down any http traffic coming into our server **192.168.250.70** containing the term ".exe." This query may not lead to the findings, but it's good to start from 1 extension and move ahead.

Search Query: `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" *.exe`

The screenshot shows the Splunk search interface. At the top, there's a navigation bar with links for Search, Datasets, Reports, Alerts, and Dashboards. On the right, there's a "Search & Reporting" button. Below the navigation is a search bar with placeholder text "enter search here...". To the right of the search bar are buttons for "All time" and a magnifying glass icon. Under the search bar, it says "No Event Sampling" with a dropdown arrow. On the far right, there's a "Smart Mode" dropdown. The main area is titled "Search" and contains sections for "How to Search" (with links to Documentation and Tutorial), "What to Search" (showing 956,462 indexed events from 5 years ago to a few seconds ago), and a "Data Summary" section. A "Search History" link is also present.

With the search query in place, we are looking for the fields that could have some values of our interest. As we could not find the file name field, we looked at the missing fields and saw a field. `part_filename{}`.

Observing the interesting fields and values, we can see the field `part_filename{}` contains the two file names. an executable file `3791.exe` and a PHP file `agent.php`

The screenshot shows the Splunk search results for the query `index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" *.exe`. It displays 17 events found before 3/5/22 12:08:34.000 AM with no event sampling. The interface includes tabs for Events (17), Patterns, Statistics, and Visualization, along with timeline controls for Format Timeline, Zoom Out, Zoom to Selection, and Deselect.

A tooltip is open over the `part_filename{}` field. The tooltip header is `part_filename{}`. It states "2 Values, 5.882% of events" and has a "Selected" button with "Yes" and "No" options. Below this, there are sections for "Reports" (Top values, Top values by time, Events with this field) and "Rare values". A table titled "Values" shows two entries:

Values	Count	%
3791.exe	1	100%
agent.php	1	100%

A red arrow points from the "INTERESTING FIELDS" list on the left towards the tooltip. The "INTERESTING FIELDS" list includes `host`, `part_filename{}`, `source`, and `sourcetype`.

Next, we need to find if any of these files came from the IP addresses that were found to be associated with the attack earlier.

Click on the file name; it will be added to the search query, then look for the field c_ip, which seems to represent the client IP.

Search Query: index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70"
"part_filename{}"="3791.exe"

The screenshot shows a Splunk search results page. The search query at the top is: index=botsv1 sourcetype=stream:http dest_ip="192.168.250.70" "part_filename{}"="3791.exe". Below the query, it says "1 event (before 3/5/22 12:15:43.000 AM) No Event Sampling". The main area shows a single event with a green bar. On the left, there's a sidebar with "SELECTED FIELDS" containing "a c_ip 1", "a host 1", "a part_filename[] 2", "a source 1", and "a sourcetype 1". A red arrow points from the "Selected" dropdown menu in this sidebar to the "Values" table on the right. The "Values" table has columns "Values", "Count", and "%". It shows one row: "40.80.148.42" with a count of 1 and 100%. There are also tabs for "Reports" (Top values, Top values by time, Events with this field) and "Rare values".

Was this file executed on the server after being uploaded?

We have found that file 3791.exe was uploaded on the server. The question that may come to our mind would be, was this file executed on the server? We need to narrow down our search query to show the logs from the host-centric log sources to answer this question.

Search Query: index=botsv1 "3791.exe"

index=botsv1 "3791.exe"

✓ 76 events (before 3/5/22 12:18:54.000 AM) No Event Sampling ▾

Events (76) Patterns Statistics Visualization

Format Timeline ▾ - Zoom Out + Zoom to Selection × Deselect

sourcetype

< Hide Fields All Fields

SELECTED FIELDS

- a host 4
- a part_filename[] 2
- a source 5
- a sourcetype 5**

INTERESTING FIELDS

- a Channel 1
- a Computer 1
- a dvc 3
- a dvc_nt_host 1
- # event_id 72
- # EventCode 6
- a EventData_Xml 67

Reports

Top values Top values by time Rare values

Events with this field

Values	Count	%
xmlwineventlog	69	90.789%
wineventlog	3	3.947%
stream:http	2	2.632%
fortigate_utm	1	1.316%
suricata	1	1.316%

Following the **Host-centric** log, sources were found to have traces of the executable 3791. exe.

- Sysmon
- WinEventlog
- fortigate_utm

For the evidence of execution, we can leverage sysmon and look at the EventCode=1 for program execution.

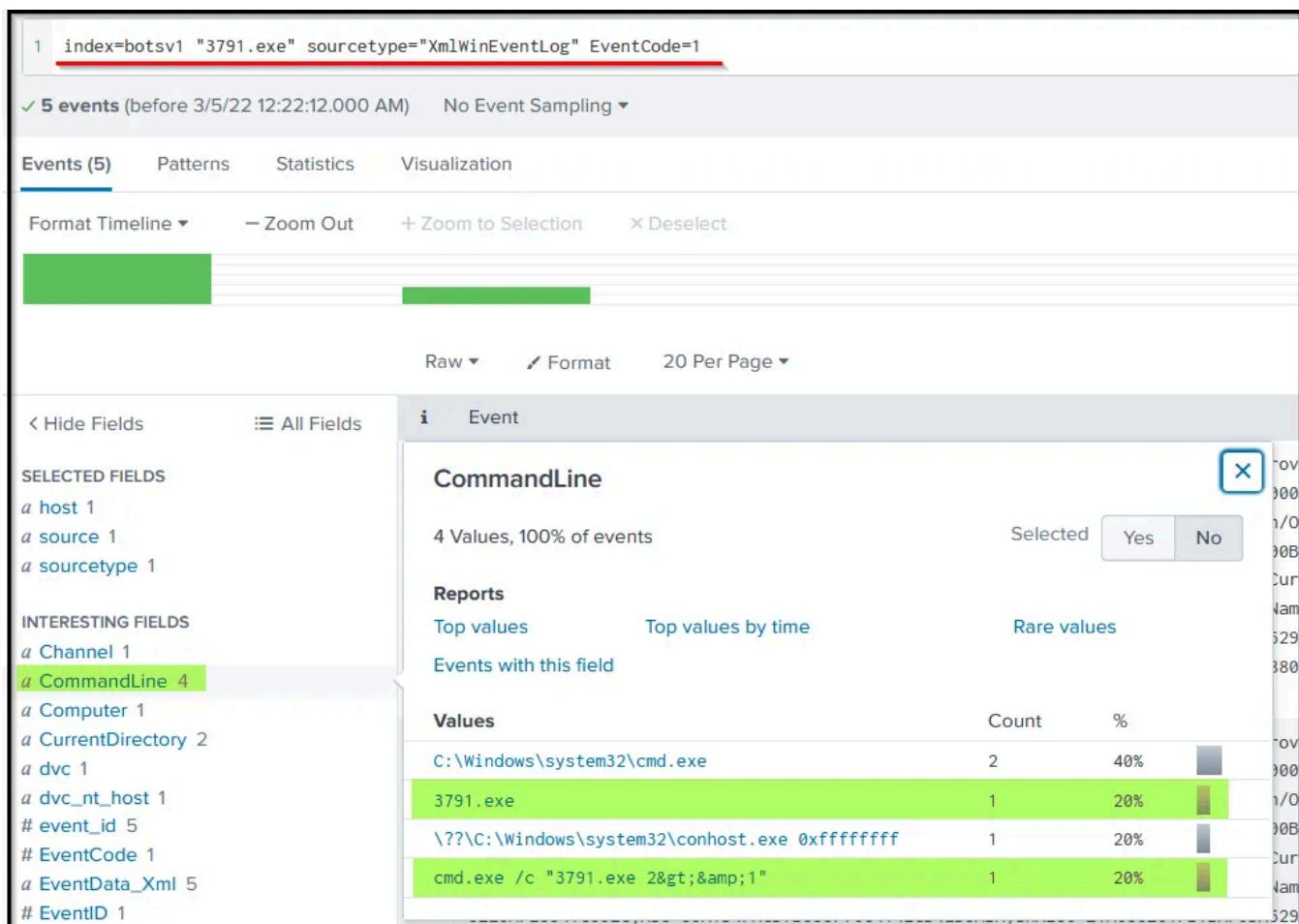
Reference: <https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon>

Event ID 1: Process creation

The process creation event provides extended information about a newly created process. The full command line provides context on the process execution. The ProcessGUID field is a unique value for this process across a domain to make event correlation easier. The hash is a full hash of the file with the algorithms in the HashType field.

Search Query: index=botsv1 "3791.exe" sourcetype="XmlWinEventLog" EventCode=1

Query Explanation: This query will look for the process Creation logs containing the term “3791.exe” in the logs.



Looking at the output, we can clearly say that this file was executed on the compromised server. We can also look at other host-centric log sources to confirm the result.

Answer the questions below :

1-Sysmon also collects the Hash value of the processes being created. What is the MD5 HASH of the program 3791.exe?

```
a dvc 1
a dvc_nt_host 1
# event_id 1
# EventCode 1
a EventData_Xml 1
# EventID 1
# EventRecordID 1
a eventtype 2
a Guid 1
a Hashes 1
# id 1
a Image 1
a Index 1
a IntegrityLevel 1
a Keywords 1
```

Hashes

1 Value, 100% of events Selected

Reports

[Top values](#) [Top values by time](#) [Rare values](#)

Events with this field

Values	Count	%
SHA1=65DF73D77324D008C83C3E57B445DF0FD43A3A51,MD5=AAE3F5A 29935E6ABCC2C2754D12A9AF0,SHA256=EC78C938D8453739CA2A370B 9C275971EC46CAF6E479DE2B2D04E97CC47FA45D,IMPHASH=481F47BB B2C9C21E108D65F52B04C448	1	100%

Answer: AAE3F5A29935E6ABCC2C2754D12A9AF0

2-Looking at the logs, which user executed the program 3791.exe on the server?

```
# ProcessId 1
a ProcessID 1
a punct 1
# RecordNumber 1
# signature_id 1
a splunk_server 1
a System_Props_Xml 1
a SystemTime 1
a tag 1
a tag:eventtype 1
# Task 1
# TerminalSessionId 1
a ThreadID 1
a User 1 ①
a user_id 1
a UserID 1
a UtcTime 1
a vendor_product 1
```

User

1 Value, 100% of events Selected

Reports

[Top values](#) [Top values by time](#) [Rare values](#)

Events with this field

Values	Count	%
NT AUTHORITY\IUSR ②	1	100%

Answer: NT AUTHORITY\IUSR

3-Search hash on the virustotal. What other name is associated with this file 3791.exe?

Names ①

ab.exe

ec78c938d8453739ca2a370b9c275971ec46caf6e479de2b2d04e97cc47fa45d
aae3f5a29935e6abcc2c2754d12a9af0.virobj
3791.exe

Answer : ab.exe

Task 7: Action on Objectives



As the website was defaced due to a successful attack by the adversary, it would be helpful to understand better what ended up on the website that caused defacement.

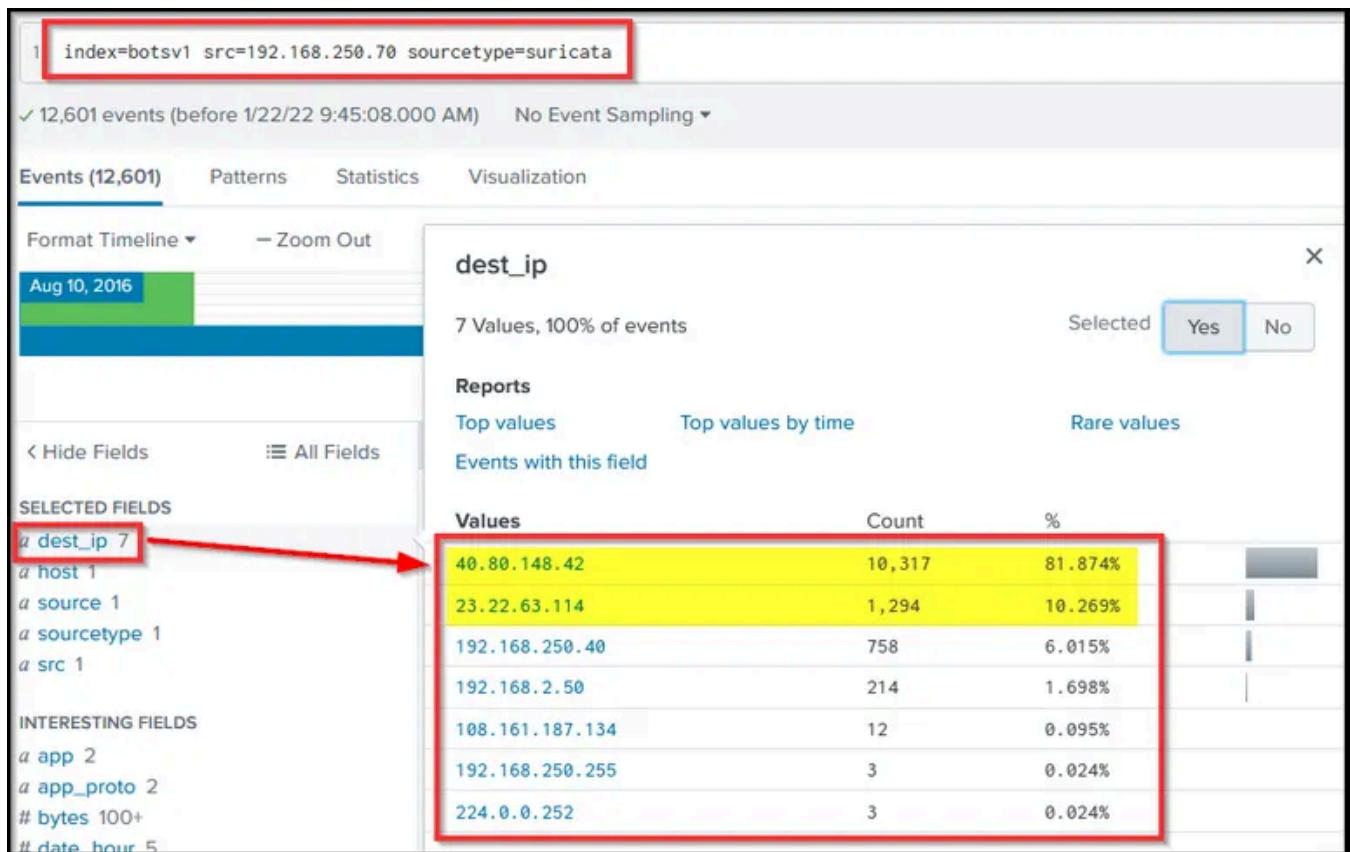
As an analyst, our first quest could be to figure out the traffic flow that could lead us to the answer to this question. There can be a different approach to finding the answer to this question. We will start our investigation by examining the **Suricata** log source and the IP addresses communicating with the webserver 192.168.250.70.

Search Query: index=botsv1 dest=192.168.250.70 sourcetype=suricata

Values	Count	%
192.168.2.50	211	50.119%
192.168.250.70	210	49.881%

The logs do not show any external IP communicating with the server. Let us change the flow direction to see if any communication originates from the server.

Search Query: index=botsv1 src=192.168.250.70 sourcetype=suricata



What is interesting about the output? Usually, the web servers do not originate the traffic. The browser or the client would be the source, and the server would be the destination. Here we see three external IPs towards which our web server initiates the outbound traffic. There is a large chunk of traffic going to these external IP addresses, which could be worth checking.

Pivot into the destination IPs one by one to see what kind of traffic/communication is being carried out.

Search Query: index=botsv1 src=192.168.250.70 sourcetype=suricata

dest_ip=23.22.63.114

index=botsv1 src=192.168.250.70 sourcetype=suricata dest_ip=23.22.63.114

1,294 events (before 1/22/22 10:11:15.000 AM) No Event Sampling ▾

Events (1,294) Patterns Statistics Visualization

Format Timeline ▾ – Zoom Out + Zoom to Selection × Deselect

List ▾ ✎ Format 20 Per Page ▾

< Hide Fields All Fields

SELECTED FIELDS

- # dest_ip 1
- # source 1
- # sourcetype 1
- # src 1
- url 3**

INTERESTING FIELDS

- # app 1
- # app_proto 1
- # bytes 20
- # date_hour 2
- # date_mday 1

url

3 Values, 99.691% of events Selected Yes No

Reports

Top values Top values by time Rare values

Events with this field

Values	Count	%
/joomla/administrator/index.php	1,235	95.736%
/joomla/agent.php	52	4.031%
/poisonivy-is-coming-for-you-batman.jpeg	3	0.232%

The URL field shows 2 PHP files and one jpeg file. This jpeg file looks interesting. Let us change the search query and see where this jpeg file came from.

Search Query: index=botsv1 url="/poisonivy-is-coming-for-you-batman.jpeg"

```
dest_ip="192.168.250.70" | table _time src dest_ip http.hostname url
```

Overview Scenario #1 - APT Scenario #2 - Ransomware Supplemental Material Search Dashboards

Investigating with Splunk Workshop

New Search

1 index=botsv1 src=192.168.250.70 sourcetype=suricata dest_ip=23.22.63.114

1,294 events (before 1/22/22 10:23:34.000 AM) No Event Sampling ▾

Events (1,294) Patterns Statistics Visualization

Format Timeline ▾ – Zoom Out + Zoom to Selection × Deselect

List ▾ ✎ Format 20 Per Page ▾

< Hide Fields All Fields

SELECTED FIELDS

- # dest_ip 1
- # source 1
- # sourcetype 1
- # src 1
- # url 3

INTERESTING FIELDS

- # app 1
- # app_proto 1
- # bytes 20
- # date_hour 2
- # date_mday 1
- # date_minute 19
- # date_month 1
- # date_second 60
- # date_wday 1
- # date_year 1
- # date_zone 1
- # dest 3

url

3 Values, 99.691% of events Selected Yes No

Reports

Top values Top values by time Rare values

Events with this field

Values	Count	%
/joomla/administrator/index.php	1,235	95.736%
/joomla/agent.php	52	4.031%
/poisonivy-is-coming-for-you-batman.jpeg	3	0.232%

Show as raw text

dest_ip = 23.22.63.114 : source = /var/log/suricata/eve.json : sourcetype = suricata : src = 192.168.250.70

> 8/10/16 10:21:34.828 PM { [] } app_proto: http dest_ip: 23.22.63.114 dest_port: 47517

The end result clearly shows a suspicious jpeg poisonivy-is-coming-for-you-batman.jpeg was downloaded from the attacker's host

prankglassinebracket.jumpingcrab.com that defaced the site.

Answer the questions below:

1-What is the name of the file that defaced the imreallynotbatman.com website ?

Answer: poisonivy-is-coming-for-you-batman.jpeg

2-Fortigate Firewall 'fortigate_utm' detected SQL attempt from the attacker's IP 40.80.148.42. What is the name of the rule that was triggered during the SQL Injection attempt?

Hint: attack field

Top 10 Values	Count	%
Acunetix.Web.Vulnerability.Scanner	4,230	94.272%
HTTP.URI.SQL.Injection	199	4.435%
Bash.Function.Definitions.Remote.Code.Execution	18	0.401%
Apache.Camel.XSLT.Component.XXE	12	0.267%
PHP.CGI.Argument.Injection	10	0.223%

Answer: HTTP.URI.SQL.Injection

Task 8: Command and Control Phase



The attacker uploaded the file to the server before defacing it. While doing so, the attacker used a Dynamic DNS to resolve a malicious IP. Our objective would be to find the IP that the attacker decided the DNS.

To investigate the communication to and from the adversary's IP addresses, we will be examining the network-centric log sources mentioned above. We will first pick `fortigate_utm` to review the firewall logs and then move on to the other log sources.

Search Query: `index=botsv1 sourcetype=fortigate_utm "poisonivy-is-coming-for-you-batman.jpeg"`

1 index=botsv1 sourcetype=fortigate_utm "poisonivy-is-coming-for-you-batman.jpeg"

✓ 3 events (before 3/5/22 1:27:00:00 AM) No Event Sampling ▾ Job ▾

Events (3) Patterns Statistics Visualization

Format Timeline ▾ — Zoom Out + Zoom to Selection X Deselect

List ▾ Format 20 Per Page ▾

Hide Fields All Fields

SELECTED FIELDS
`a_dest_ip` 1
`a_host` 1
`a_source` 1
`a_sourcetype` 1
`a_src` 1
`a_url` 1

INTERESTING FIELDS
`a_action` 1
`a_app` 1
`#bytes` 1
`#bytes_in` 1
`#bytes_out` 1
`#cat` 1
`a_catdesc` 1
`a_category` 1
`a_crlvl` 1
`#crscore` 1
`a_date` 1

i Time	Event
> 8/10/16 10:19:10.000 PM	Aug 10 16:19:10 192.168.250.1 date=2016-08-10 time=16:19:10 devname=gotham-fortigate devid=FGT60D4614044725 logid=0317013312 type=utm sub ice vd="root" policyid=10 sessionid=932526 user="" srcip=192.168.250.70 srport=51573 srcinf="internal3" dstip=23.22.63.114 dstport=1337 bracket.jumpingcrab.com:1337" profile="monitor-all" action="passthrough retype=direct url="/poisonivy-is-coming-for-you-batman.jpeg" sent longs to an allowed category in policy" method=domain cat=26 catdesc="Malicious Websites" crscore=30 crlevel=high dest_ip = 23.22.63.114 host = 192.168.250.1 source = udp:514 sourcetype = fortigate_utm src = 192.168.250.70 url = prankglassinebracket.jumpingcrab.com:1337/poisonivy-is-coming-for-you-batma...
> 8/10/16 10:13:46.000 PM	Aug 10 16:13:46 192.168.250.1 date=2016-08-10 time=16:13:46 devname=gotham-fortigate devid=FGT60D4614044725 logid=0317013312 type=utm sub ice vd="root" policyid=10 sessionid=930693 user="" srcip=192.168.250.70 srport=63139 srcinf="internal3" dstip=23.22.63.114 dstport=1337 bracket.jumpingcrab.com:1337" profile="monitor-all" action="passthrough retype=direct url="/poisonivy-is-coming-for-you-batman.jpeg" sent longs to an allowed category in policy" method=domain cat=26 catdesc="Malicious Websites" crscore=30 crlevel=high dest_ip = 23.22.63.114 host = 192.168.250.1 source = udp:514 sourcetype = fortigate_utm src = 192.168.250.70 url = prankglassinebracket.jumpingcrab.com:1337/poisonivy-is-coming-for-you-batma...
> 8/10/16 10:06:21.000 PM	Aug 10 16:06:21 192.168.250.1 date=2016-08-10 time=16:06:21 devname=gotham-fortigate devid=FGT60D4614044725 logid=0317013312 type=utm sub ice vd="root" policyid=10 sessionid=928318 user="" srcip=192.168.250.70 srport=56504 srcinf="internal3" dstip=23.22.63.114 dstport=1337 bracket.jumpingcrab.com:1337" profile="monitor-all" action="passthrough retype=direct url="/poisonivy-is-coming-for-you-batman.jpeg" sent longs to an allowed category in policy" method=domain cat=26 catdesc="Malicious Websites" crscore=30 crlevel=high dest_ip = 23.22.63.114 host = 192.168.250.1 source = udp:514 sourcetype = fortigate_utm src = 192.168.250.70 url = prankglassinebracket.jumpingcrab.com:1337/poisonivy-is-coming-for-you-batma...

Looking into the Fortinet firewall logs, we can see the src IP, destination IP, and URL. Look at the fields on the left panel and the field `url` contains the FQDN (Fully Qualified Domain Name).

Selected Fields: `a source 1`, `a sourcetype 1`, `a src 1`, `a url 1`

INTERESTING FIELDS: `a action 1`, `# bytes 1`, `# bytes_in 1`, `# bytes_out 1`

url

1 Value, 100% of events

Reports: Top values, Top values by time, Rare values

Events with this field

Values	Count	%
<code>prankglassinebracket.jumpingcrab.com:1337/poisonivy-is-coming-for-you-batman.jpeg</code>	3	100%

Though we have found the answer, we can verify other log sources.

Let us verify the answer by looking at another log source. `stream:http`.

Search Query: `index=botsv1 sourcetype=stream:http dest_ip=23.22.63.114 "poisonivy-is-coming-for-you-batman.jpeg" src_ip=192.168.250.70`

```

dest_ip: 23.22.63.114
dest_mac: 08:5B:0E:93:92:AF
dest_port: 1337
duplicate_packets_in: 2
duplicate_packets_out: 0
endtime: 2016-08-10T22:13:46.915172Z
http_method: GET
missing_packets_in: 0
missing_packets_out: 0
network_interface: eth1
packets_in: 6
packets_out: 5
reply_time: 0
request: GET /poisonivy-is-coming-for-you-batman.jpeg HTTP/1.0
request_ack_time: 3246
request_time: 61714
response_ack_time: 0
response_time: 0
server_rtt: 32357
server_rtt_packets: 2
server_rtt_sum: 64714
site: prankglassinebracket.jumpingcrab.com:1337
src_headers: GET /poisonivy-is-coming-for-you-batman.jpeg HTTP/1.0
Host: prankglassinebracket.jumpingcrab.com:1337

src_ip: 192.168.250.70
src_mac: 00:0C:29:C4:02:7E
src_port: 63139
time_taken: 61715
timestamp: 2016-08-10T22:13:46.853458Z
transport: tcp
uri: /poisonivy-is-coming-for-you-batman.jpeg
uri_path: /poisonivy-is-coming-for-you-batman.jpeg

```

We have identified the suspicious domain as a Command and Control Server, which the attacker contacted after gaining control of the server.

Note: We can also confirm the domain by looking at the last log source `stream:dns` to see what DNS queries were sent from the webserver during the infection period.

Answer the questions below:

1-This attack used dynamic DNS to resolve to the malicious IP. What fully qualified domain name (FQDN) is associated with this attack?

Answer : *prankglassinebracket.jumpingcrab.com*

Task 9: Weaponization Phase



Weaponization

In the weaponization phase, the adversaries would:

- Create Malware / Malicious document to gain initial access / evade detection etc.
- Establish domains similar to the target domain to trick users.
- Create a Command and Control Server for the post-exploitation communication/activity etc.

We have found some domains / IP addresses associated with the attacker during the investigations. This task will mainly look into OSINT sites to see what more information we can get about the adversary.

So far, we have found a domain `prankglassinebracket.jumpingcrab.com` associated with this attack. Our first task would be to find the IP address tied to the domains that may potentially be pre-staged to attack Wayne Enterprise.

In the following exercise, we will be searching the online Threat Intel sites for any information like IP addresses/domains / Email addresses associated with this domain which could help us know more about this adversary.

Robtex:

Robtex is a Threat Intel site that provides information about IP addresses, domain names, etc.

Please search for the domain on the robtex site and see what we get. We will get the IP addresses associated with this domain.

The screenshot shows a browser window displaying the Robtex website at `https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com`. The page has a green header bar with the title "ANALYSIS". Below it, a sub-section titled "Results found" lists "Jumpingcrab.com.". The main content area is titled "QUICK INFO" and contains a table with the following data:

General	
FQDN	prankglassinebracket.jumpingcrab.com
Host Name	prankglassinebracket
Domain Name	jumpingcrab.com
Registry	com
TLD	com
Domain DNS	
Name servers	ns1.afraid.org ns2.afraid.org ns3.afraid.org ns4.afraid.org
Mail servers	mail.jumpingcrab.com
IP Numbers	69.197.18.183 70.39.97.227 169.47.130.85

Some domains/subdomains associated with this domain:

SHARED

This section shows related hostnames and ipnumbers

Siblings

Siblings are domains or hostnames on the same level, under the same parent level. Not necessarily related in any other way

adjazd.jumpingcrab.com
bfffbff.jumpingcrab.com
hwd.jumpingcrab.com
mclpcb.jumpingcrab.com
nonesuch.jumpingcrab.com
piranhabrothers.jumpingcrab.com
sendmgs.jumpingcrab.com
sslsls.jumpingcrab.com
www.jumpingcrab.com
zim.jumpingcrab.com

10 results shown.

Reference: <https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com>

Next, search for the IP address 23.22.63.114 on this Threat Intel site.

ANALYSIS

This section shows a quick analysis of the given host name or ip number.

23.22.63.114 has one PTR.

PTR

The PTR is [ec2-23-22-63-114.compute-1.amazonaws.com](https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com). The IP number is in Ashburn, United States. It is hosted by Amazon EC2 IAD prefix.

We investigated eight host names that point to 23.22.63.114. Example: [ec2-23-22-63-114.compute-1.amazonaws.com](https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com), [waynecrpinc.com](https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com), [waynecorpnc.com](https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com) and [wanecorpinc.com](https://www.robtex.com/dns-lookup/prankglassinebracket.jumpingcrab.com).

What did we find? this IP is associated with some domains that look pretty similar to the WAYNE Enterprise site.

Reference: <https://www.robtex.com/ip-lookup/23.22.63.114>

SHARED

This section shows related hostnames and ipnumbers

Using as IP number

wanecorpinc.com
 wayncorpinc.com
 waynecorinc.com
 waynecorpnc.com
 waynecrpinc.com
 wayneorpinc.com
 wynecorpinc.com
 ec2-23-22-63-114.compute-1.amazonaws.com

8 results shown.

<

Virustotal:

Virustotal is an OSINT site used to analyze suspicious files, domains, IP, etc. Let's now search for the IP address on the virustotal site. If we go to the **RELATIONS** tab, we can see all the domains associated with this IP which look similar to the Wayne Enterprise company.

23.22.63.114

Did you intend to search across the file corpus instead? [Click here](#)

0 / 90

Community Score

2 detected files communicating with this IP address

23.22.63.114 (23.20.0.0/14)
AS 14618 (AMAZON-AES)

DETECTION DETAILS RELATIONS COMMUNITY 5

Passive DNS Replication

Date resolved	Detections	Resolver	Domain
2019-12-01	0 / 89	VirusTotal	waynecorinc.com
2019-11-30	0 / 89	VirusTotal	wanecorpinc.com
2019-11-29	0 / 89	VirusTotal	wynecorpinc.com
2019-11-28	0 / 89	VirusTotal	wayneorpinc.com
2019-11-05	0 / 89	VirusTotal	wynecorpinc.com
2019-09-30	0 / 89	VirusTotal	wayneorpinc.com
2019-09-28	0 / 89	VirusTotal	waynecorpinc.com
2019-04-19	0 / 89	VirusTotal	ec2-23-22-63-114.compute-1.amazonaws.com
2018-07-18	0 / 90	VirusTotal	po1s0n1vy.com
2018-05-19	0 / 90	VirusTotal	www.po1s0n1vy.com

Communicating Files

Scanned	Detections	Type	Name
2021-09-07	51 / 68	Win32 EXE	check.exe
2021-12-08	60 / 67	Win32 EXE	ab.exe

In the domain list, we saw the domain that is associated with the attacker www.po1s0n1vy.com. Let us search for this domain on the virustotal.

The screenshot shows a domain analysis interface for `www.po1s0n1vy.com`. At the top, there's a green circle with a '0' and a note: "No security vendors flagged this domain as malicious". Below this, the domain name is listed along with its registrar (GoDaddy.com, LLC) and creation date (1 year ago). A "Community Score" section is present.

Passive DNS Replication

Date resolved	Detections	Resolver	IP
2021-09-03	2 / 90	VirusTotal	34.102.136.180
2018-08-30	1 / 90	VirusTotal	91.195.240.117
2018-05-19	0 / 90	VirusTotal	23.22.63.114

Siblings

smtp.po1s0n1vy.com	0 / 89	91.195.240.117	64.29.151.235
Elian.po1s0n1vy.com	0 / 89	64.29.151.221	
ftp.po1s0n1vy.com	0 / 89	64.29.151.221	
illian.po1s0n1vy.com	0 / 89	64.29.151.221	
prankglassinebracket.jumpingcrab.po1s0n1vy.com	0 / 90	91.195.240.117	64.29.151.221

We can also look for the whois information on this site -> [whois.domaintools.com](https://whois.domaintools.com/po1s0n1vy.com) to see if we can find something valuable.

The screenshot shows the Whois Record for `Po1s0n1Vy.com` on Domaintools. The record includes the following details:

- Domain Profile**
 - Registrant: Registration Private
 - Registrant Org: Domains By Proxy, LLC
 - Registrant Country: us
 - Registrar: GoDaddy.com, LLC
IANA ID: 146
URL: <https://www.godaddy.com>, <http://www.godaddy.com>
Whois Server: whois.godaddy.com
abuse@godaddy.com
(p) 14806242505
 - Registrar Status: clientDeleteProhibited, clientRenewProhibited, clientTransferProhibited, clientUpdateProhibited
 - Dates: 718 days old
Created on 2020-01-09
Expires on 2022-01-09
Updated on 2021-01-10
 - Name Servers: NS37.DOMAINCONTROL.COM (has 60,029,130 domains)
NS38.DOMAINCONTROL.COM (has 60,029,130 domains)
 - Tech Contact: Registration Private
Domains By Proxy, LLC
DomainsByProxy.com,
Tempe, Arizona, 85284, us
po1s0n1vy.com@domainsbyproxy.com
(p) 14806242599 (f) 14806242598
 - IP Address: 34.102.136.180 - 29,054,317 other sites hosted on this server
 - IP Location: Missouri - Kansas City - Google
 - ASN: AS15169 GOOGLE, US (registered Mar 30, 2000)

Answer the questions below:

1-What IP address has P01s0n1vy tied to domains that are pre-staged to attack Wayne Enterprises?

23.22.63.114

0 / 87

Community Score

3 detected files communicating with this IP address

23.22.63.114 (23.20.0.0/14)
AS 14618 (AMAZON-AES)

DETECTION DETAILS RELATIONS COMMUNITY 11

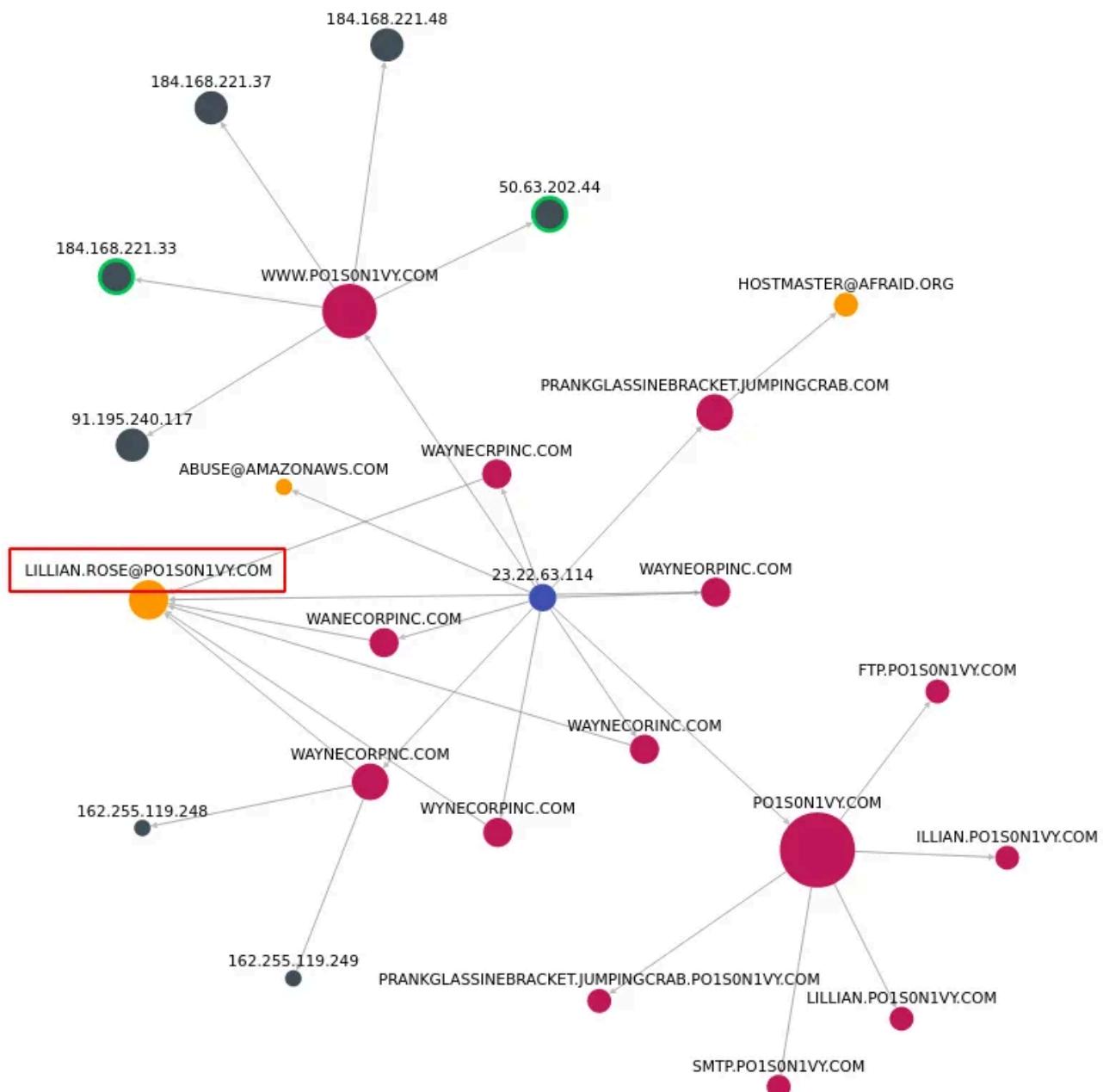
[Join the VT Community](#) and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

Date resolved	Detections	Resolver	Domain
2019-12-01	0 / 87	VirusTotal	waynecorinc.com
2019-11-30	0 / 87	VirusTotal	wanecorpinc.com
2019-11-29	0 / 87	VirusTotal	wynecorpinc.com
2019-11-28	0 / 87	VirusTotal	wayneorpinc.com
2019-11-05	0 / 87	VirusTotal	wayncorpinc.com
2019-09-30	0 / 87	VirusTotal	waynecrpinc.com
2019-09-28	0 / 87	VirusTotal	waynecorpnc.com
2019-04-19	0 / 86	VirusTotal	ec2-23-22-63-114.compute-1.amazonaws.com
2018-07-18	0 / 87	VirusTotal	po1s0n1vy.com
2018-05-19	0 / 87	VirusTotal	www.po1s0n1vy.com

Answer: 23.22.63.114

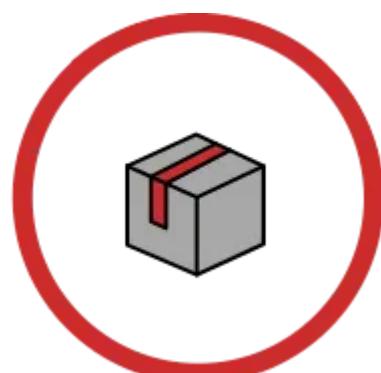
2-Based on the data gathered from this attack and common open-source intelligence sources for domain names, what is the email address that is most likely associated with the P01s0n1vy APT group?

Googling for the IOCs collected so far leads to <https://threatcrowd.org/ip.php?ip=23.22.63.114> where we are presented with a relationship diagram involving domains, IPs, emails:



Answer: `lillian.rose@po1s0nvy.com`

Task 10: Delivery Phase



Delivery

Attackers create malware and infect devices to gain initial access or evade defenses and find ways to deliver it through different means. We have identified various IP addresses, domains and Email addresses associated with this adversary. Our task for this lesson would be to use the information we have about the adversary and use various Threat Hunting platforms and OSINT sites to find any malware linked with the adversary.

Threat Intel report suggested that this adversary group Poison Ivy appears to have a secondary attack vector in case the `initial compromise` fails. Our objective would be to understand more about the attacker and their methodology and correlate the information found in the logs with various threat Intel sources.

OSINT sites

- Virustotal
- ThreatMiner
- Hybrid-Analysis

ThreatMiner

Let's start our investigation by looking for the IP `23.22.63.114` on the Threat Intel site [ThreatMiner](#).

The screenshot shows a ThreatMiner search results page. At the top, there are buttons for Copy, Excel, CSV, and PDF. Below is a table with two columns: MD5 and Detections.

MD5	Detections																																								
aae3f5a29935e6abcc2c2754d12a9af0	N/A																																								
39eecefa9a13293a93bb20036eaf1f5e	N/A																																								
c99131e0169171935c5ac32615ed6261	<table border="1"> <tr><td>ALYac</td><td>Trojan.GenericKD.3470547</td></tr> <tr><td>AVG</td><td>Agent5.APHV</td></tr> <tr><td>AVware</td><td>Trojan.Win32.Generic!IBT</td></tr> <tr><td>Ad-Aware</td><td>Trojan.GenericKD.3470547</td></tr> <tr><td>AegisLab</td><td>Agent5.Aphv.Gen!C</td></tr> <tr><td>AhnLab-V3</td><td>Malware/Gen.Generic.N2081883700</td></tr> <tr><td>Anti-AVL</td><td>Trojan[Backdoor]Win32.Redspip</td></tr> <tr><td>Arcabit</td><td>Trojan.Generic.D34F4D3</td></tr> <tr><td>Avira</td><td>TR/AD.Zupdax.qmyx</td></tr> <tr><td>BitDefender</td><td>Trojan.GenericKD.3470547</td></tr> <tr><td>DrWeb</td><td>Trojan.MulDrop6.51432</td></tr> <tr><td>ESET-NOD32</td><td>a variant of Win32/Korplug.HP</td></tr> <tr><td>Emsisoft</td><td>Trojan.GenericKD.3470547 (B)</td></tr> <tr><td>F-Secure</td><td>Trojan.GenericKD.3470547</td></tr> <tr><td>Fortinet</td><td>W32/Korplug.HPItr</td></tr> <tr><td>GData</td><td>Trojan.GenericKD.3470547</td></tr> <tr><td>Ikarus</td><td>Trojan.Win32.Korplug</td></tr> <tr><td>Jiangmin</td><td>Backdoor.Redsip.f</td></tr> <tr><td>K7AntiVirus</td><td>Trojan (004fc211)</td></tr> <tr><td>K7GW</td><td>Trojan (004fc211)</td></tr> </table>	ALYac	Trojan.GenericKD.3470547	AVG	Agent5.APHV	AVware	Trojan.Win32.Generic!IBT	Ad-Aware	Trojan.GenericKD.3470547	AegisLab	Agent5.Aphv.Gen!C	AhnLab-V3	Malware/Gen.Generic.N2081883700	Anti-AVL	Trojan[Backdoor]Win32.Redspip	Arcabit	Trojan.Generic.D34F4D3	Avira	TR/AD.Zupdax.qmyx	BitDefender	Trojan.GenericKD.3470547	DrWeb	Trojan.MulDrop6.51432	ESET-NOD32	a variant of Win32/Korplug.HP	Emsisoft	Trojan.GenericKD.3470547 (B)	F-Secure	Trojan.GenericKD.3470547	Fortinet	W32/Korplug.HPItr	GData	Trojan.GenericKD.3470547	Ikarus	Trojan.Win32.Korplug	Jiangmin	Backdoor.Redsip.f	K7AntiVirus	Trojan (004fc211)	K7GW	Trojan (004fc211)
ALYac	Trojan.GenericKD.3470547																																								
AVG	Agent5.APHV																																								
AVware	Trojan.Win32.Generic!IBT																																								
Ad-Aware	Trojan.GenericKD.3470547																																								
AegisLab	Agent5.Aphv.Gen!C																																								
AhnLab-V3	Malware/Gen.Generic.N2081883700																																								
Anti-AVL	Trojan[Backdoor]Win32.Redspip																																								
Arcabit	Trojan.Generic.D34F4D3																																								
Avira	TR/AD.Zupdax.qmyx																																								
BitDefender	Trojan.GenericKD.3470547																																								
DrWeb	Trojan.MulDrop6.51432																																								
ESET-NOD32	a variant of Win32/Korplug.HP																																								
Emsisoft	Trojan.GenericKD.3470547 (B)																																								
F-Secure	Trojan.GenericKD.3470547																																								
Fortinet	W32/Korplug.HPItr																																								
GData	Trojan.GenericKD.3470547																																								
Ikarus	Trojan.Win32.Korplug																																								
Jiangmin	Backdoor.Redsip.f																																								
K7AntiVirus	Trojan (004fc211)																																								
K7GW	Trojan (004fc211)																																								

We found three files associated with this IP, from which one file with the hash value c99131e0169171935c5ac32615ed6261 seems to be malicious and something of interest.

Now, click on this MD5 hash value to see the metadata and other important information about this particular file.

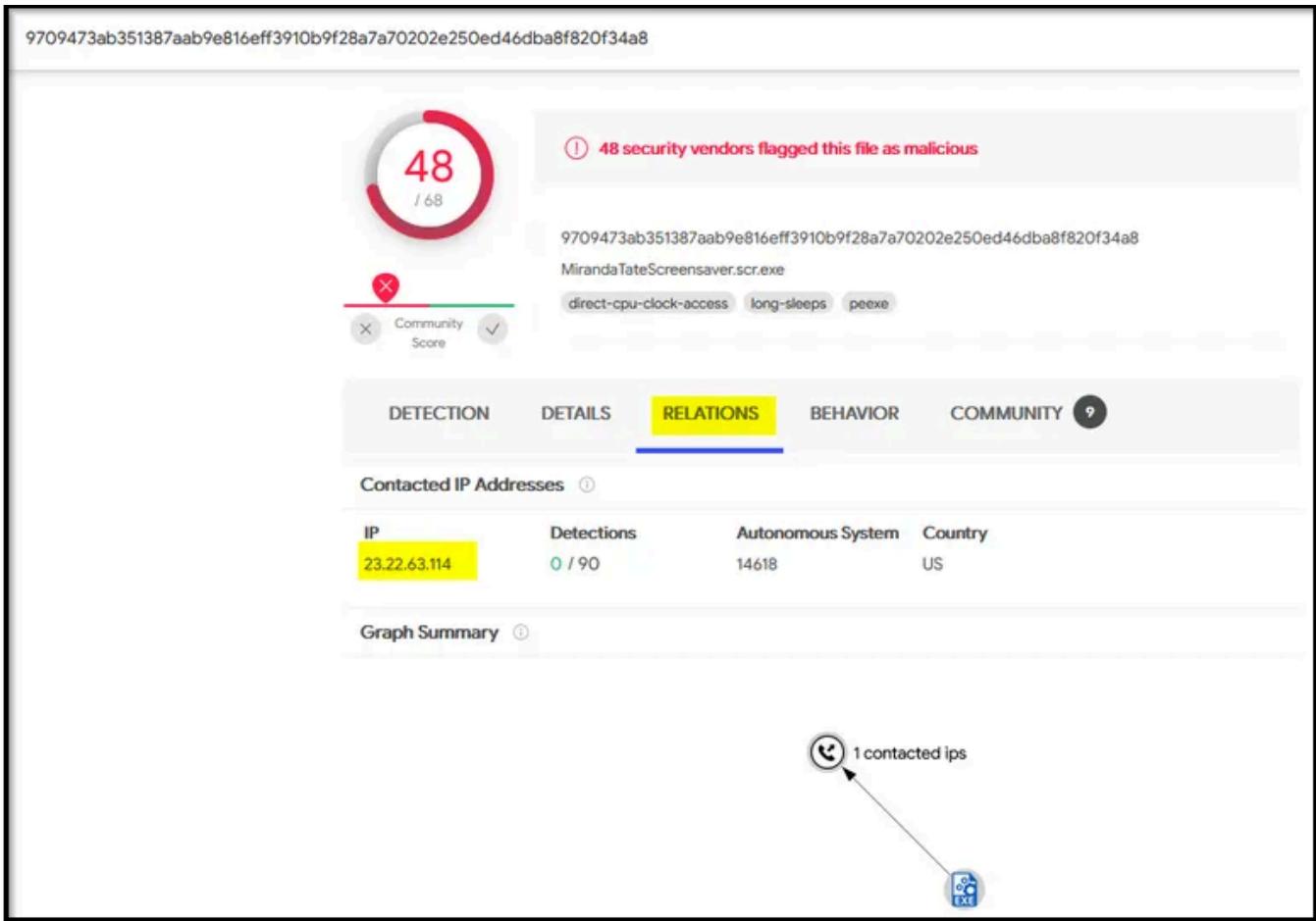
The screenshot shows the detailed metadata for the file with MD5 hash c99131e0169171935c5ac32615ed6261. The file name is MirandaTateScreensaver.scr.exe. The analysis date is 2016-09-01 09:03:44. The SHA256 hash is 9709473ab351387aab9be816eff3910b9f28a7a70202e250ed46dba8f820f34a8. The file type is PE32 executable (console) Intel 80386, for MS Windows. The file size is 494080 bytes. The MD5 hash is c99131e0169171935c5ac32615ed6261. The SHA1 hash is bc927ff06263351f43db8dec88e4b08485e07996. The SHA512 hash is 8fb3b09541b021e06eee455876526607114adb547eacb7556d578c08959154b80f01bac905383a5eb4c8a9091a3fb14dc13badc36a05ea7718bf4b1053f2fdb. The SSDeep hash is 12288.JCy+DdcUyY4lO3Rc5F5H8q3iHsRaZ0 J+COpiO3Rc5F5H8q3/yaRaZ0. The IMPHASH is fae2c8486a11f609323cc15c0ee838cf. The Authentihash is N/A. The Related resources section contains links to VirusTotal, Hybrid-Analysis, and VirusShare.

Reference: <https://www.threatminer.org/host.php?q=23.22.63.114#gsc.tab=0&gsc.q=23.22.63.114&gsc.page=1>

Virustotal

Open virustotal.com and search for the hash on the virustotal now. Here, we can get information about the metadata about this Malware in the Details tab.

Vendor	Detection	Description
Ad-Aware	Gen:Variant.Doina.t7144	AhnLab-V3
Alibaba	Backdoor:Win32/Zupdax.4fc05470	ALYac
Antiy-Avi	Trojan[Backdoor]Win32.Redslip	Arcabit
Avast	Win32:Malware-gen	AVG
Avira (no cloud)	TRIAD.Zupdax.qmyx	BitDefender
BitDefenderTheta	Gen>NN2exxF34266.EjW@wuHyqel	Comodo
CrowdStrike Falcon	WinMalicious_confidence_100% (W)	Cybereason
Cylance	Unsafe	Cynet
DrWeb	Trojan.MulDrop6.51432	Emsisoft
eScan	Gen:Variant.Doina.t7144	ESET-NOD32
F-Secure	Trojan.TRIAD.Zupdax.qmyx	FireEye
Fortinet	W32/Korplug.HP	QData



Reference:

<https://www.virustotal.com/gui/file/9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8/community>

Hybrid-Analysis

Hybrid Analysis is a beneficial site that shows the behavior Analysis of any malware. Here you can look at all the activities performed by this Malware after being executed. Some of the information that Hybrid-Analysis provides are:

- Network Communication.
- DNS Requests
- Contacted Hosts with Country Mapping
- Strings
- MITRE ATT&CK Mapping

- Malicious Indicators.
- DLLs Imports / Exports
- Mutex Information if created
- File Metadata
- Screenshots

MirandaTateScreensaver.scr.exe

This report is generated from a file or URL submitted to this webservice on November 14th 2021 23:29:13 (UTC)

Guest System: Windows 7 32 bit, Professional, 6.1 (build 7601), Service Pack 1

Report generated by Falcon Sandbox v8.49.7 © Hybrid Analysis

Threat Score: 100/100
AV Detection: 77%
Labeled as: Doina.Generic

🔗 Link 🔍 Twitter 📧 E-Mail

Incident Response

Risk Assessment

Network Behavior Contacts 1 host.

MITRE ATT&CK™ Techniques Detection

This report has 7 indicators that were mapped to 7 attack techniques and 6 tactics.

Indicators

Not all malicious and suspicious indicators are displayed. Get your own cloud service or the full version to view all details.

Scroll down, and you will get a lot of information about this Malware.

File Details

MirandaTateScreensaver.scr.exe

Filename	MirandaTateScreensaver.scr.exe
Size	483KiB (494080 bytes)
Type	pe32 executable (console) Intel 80386, for MS Windows
Description	PE32 executable (console) Intel 80386, for MS Windows
Architecture	WINDOWS
SHA256	9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8
MD5	c99131e0169171935c5ac32615ed6261
SHA1	bc927ff06263351f43db8dec88e4b08485e07996
ssdeep	12288:jCy+DdcUrY4tO3Rc5F5H8q3/HSaRanZO:ji+COpO3Rc5F5H8q3/yaRaZO
imphash	fae2c8486a11f609323cc15c0ee838cf
authentihash	7bbd34f484cace4d0d5655435b6a586f49a49cb6efc63c2f3f881b7039165c18
Compiler/Packer	VC8 -> Microsoft Corporation
PDB Timestamp	05/25/2016 07:39:35 (UTC)
PDB Pathway	
PDB GUID	00000000000000000000000000000000
Resources	
Language	ENGLISH
Icon	
Visualization	
Input File (PortEx)	
Classification (TrID)	
<ul style="list-style-type: none"> 41.0% (.EXE) Win32 Executable MS Visual C++ (generic) 36.3% (.EXE) Win64 Executable (generic) 8.6% (.DLL) Win32 Dynamic Link Library (generic) 5.9% (.EXE) Win32 Executable (generic) 2.6% (.EXE) OS/2 Executable (generic) 	

Reference: <https://www.hybrid-analysis.com/sample/9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8?environmentId=100>

Answer the questions below:

1-What is the HASH of the Malware associated with the APT group?

Metadata	
File name:	MirandaTateScreensaver.scr.exe
File type:	PE32 executable (console) Intel 80386, for MS Windows
File size:	494080 bytes
Analysis date:	2016-09-01 09:03:44
MD5:	c99131e0169171935c5ac32615ed6261

Answer: c99131e0169171935c5ac32615ed6261

2-What is the name of the Malware associated with the Poison Ivy Infrastructure?

Metadata

File name:	MirandaTateScreensaver.scr.exe
File type:	PE32 executable (console) Intel 80386, for MS Windows
File size:	494080 bytes
Analysis date:	2016-09-01 09:03:44
MD5:	c99131e0169171935c5ac32615ed6261

Answer: MirandaTateScreensaver.scr.exe

Task 11: Conclusion

Conclusion:

In this fun exercise, as a SOC Analyst, we have investigated a cyber-attack where the attacker had defaced a website ‘imreallynotbatman.com’ of the Wayne Enterprise. We mapped the attacker’s activities into the 7 phases of the Cyber Kill Chain. Let us recap everything we have found so far:

Reconnaissance Phase:

We first looked at any reconnaissance activity from the attacker to identify the IP address and other details about the adversary.

Findings:

- IP Address 40.80.148.42 was found to be scanning our webserver.
- The attacker was using Acunetix as a web scanner.

Exploitation Phase:

We then looked into the traces of exploitation attempts and found brute-force attacks against our server, which were successful.

Findings:

- Brute force attack originated from IP 23.22.63.114.
- The IP address used to gain access: 40.80.148.42
- 142 unique brute force attempts were made against the server, out of which one attempt was successful

Installation Phase:

Next, we looked at the installation phase to see any executable from the attacker's IP Address uploaded to our server.

Findings:

- A malicious executable file `3791.exe` was observed to be uploaded by the attacker.
- We looked at the sysmon logs and found the MD5 hash of the file.

Action on Objective:

After compromising the web server, the attacker defaced the website.

Findings:

- We examined the logs and found the file name used to deface the webserver.

Weaponization Phase:

We used various threat Intel platforms to find the attacker's infrastructure based on the following information we saw in the above activities.

Information we had:

Domain: `prankglassinebracket.jumpingcrab.com`

IP Address: `23.22.63.114`

Findings:

- Multiple masquerading domains were found associated with the attacker's IPs.
- An email of the user `Lillian.rose@po1s0n1vy.com` was also found associated with the attacker's IP address.

Deliver Phase:

In this phase, we again leveraged online Threat Intel sites to find malware associated with the adversary's IP address, which appeared to be a secondary attack vector if the initial compromise failed.

Findings:

- A malware name `MirandaTateScreensaver.scr.exe` was found associated with the adversary.
- MD5 of the malware was `c99131e0169171935c5ac32615ed6261`

Cybersecurity

Investigation

Tryhackme



Following

Written by Mohamed Ashraf

92 Followers · 6 Following

Penetration tester && Jr Security Analyst | SOC L1 | studied {OSCP|CCRTA|CRTO|CAP} | Top 1% on TryHackMe

No responses yet



What are your thoughts?

Respond

More from Mohamed Ashraf



Mohamed Ashraf

Attacking Web Applications with Ffuf | Skills Assessment—Walkthrough

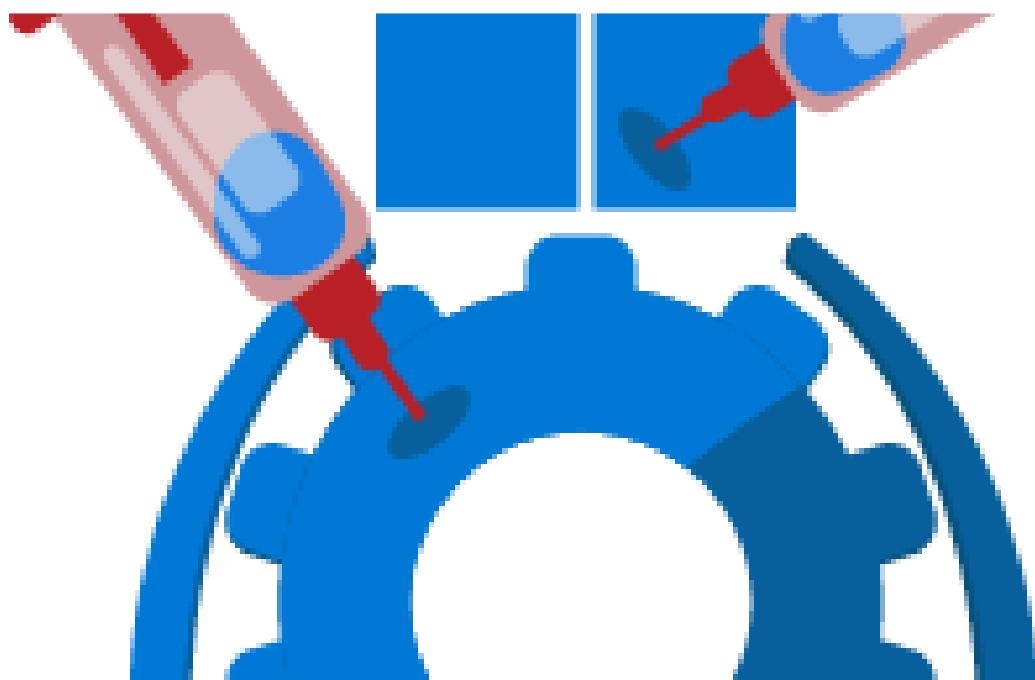
Hello friend, how are you? I hope you are well.

May 31, 2024

9



...



Mohamed Ashraf

TryHackMe| Abusing Windows Internals

Task 1: Introduction

May 7, 2023

7



...



 Mohamed Ashraf

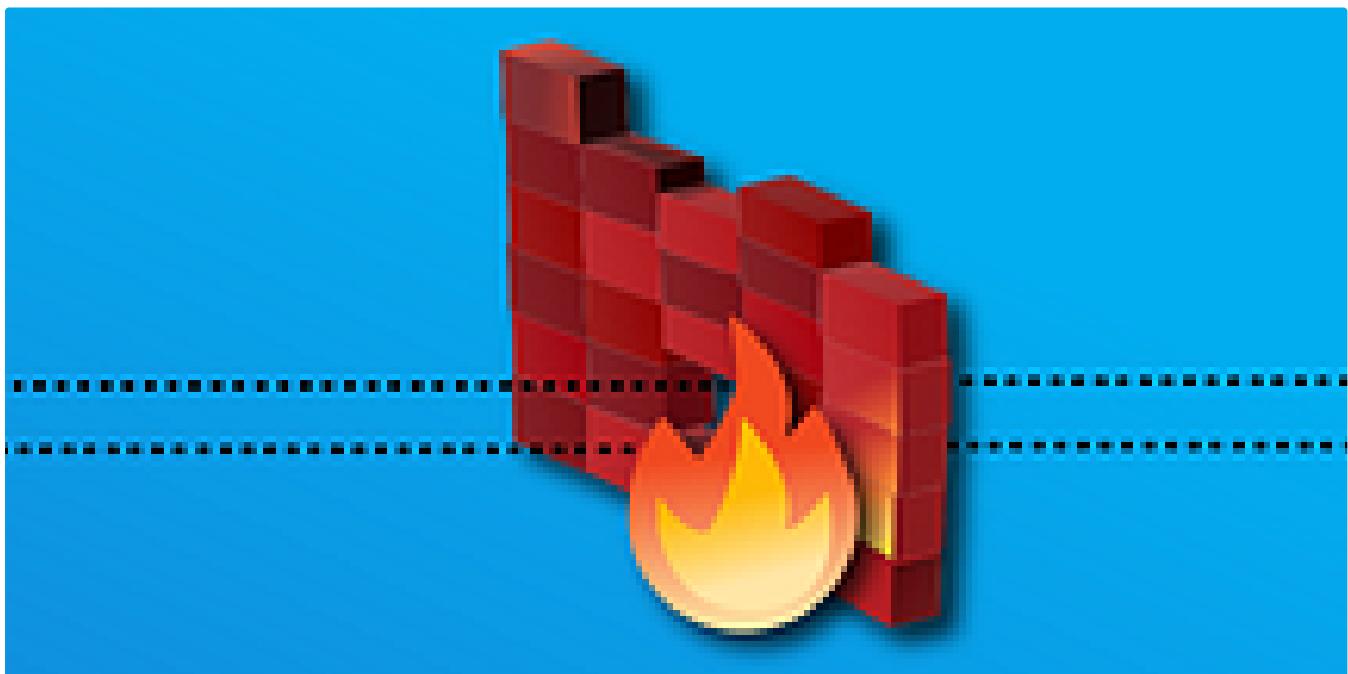
socat | Cheat Sheet

By MxOo14

May 25, 2023  2



...



 Mohamed Ashraf

TryHackMe | ItsyBitsy [Writeup]

Difficulty: Medium

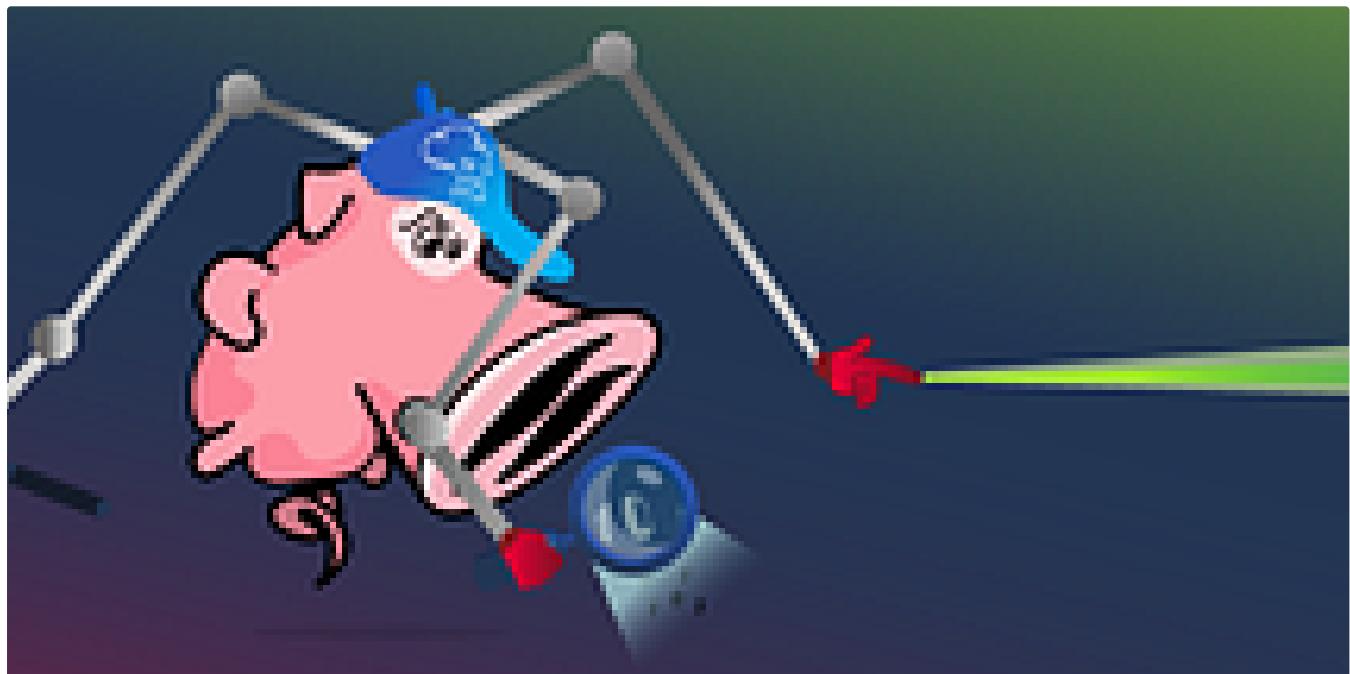
Apr 29, 2023



...

[See all from Mohamed Ashraf](#)

Recommended from Medium



In T3CH by Axoloth

TryHackMe | Snort Challenge—The Basics | WriteUp

Put your snort skills into practice and write snort rules to analyse live capture network traffic

Nov 9, 2024 100



...

Advent of Cyber 2024

Dive into the wonderful world of cyber security by engaging in festive beginner-friendly exercises every day in the lead-up to Christmas!

If you'd like to WPA, press the star key!



Day 11 Answers

cyberw1ng.medium.com

In System Weakness by Karthikeyan Nagaraj

Advent of Cyber 2024 [Day 11] Writeup with Answers | TryHackMe Walkthrough

If you'd like to WPA, press the star key!

Dec 11, 2024 855 1



...

Lists



Tech & Tools

22 stories · 387 saves



Medium's Huge List of Publications Accepting Submissions

414 stories · 4387 saves



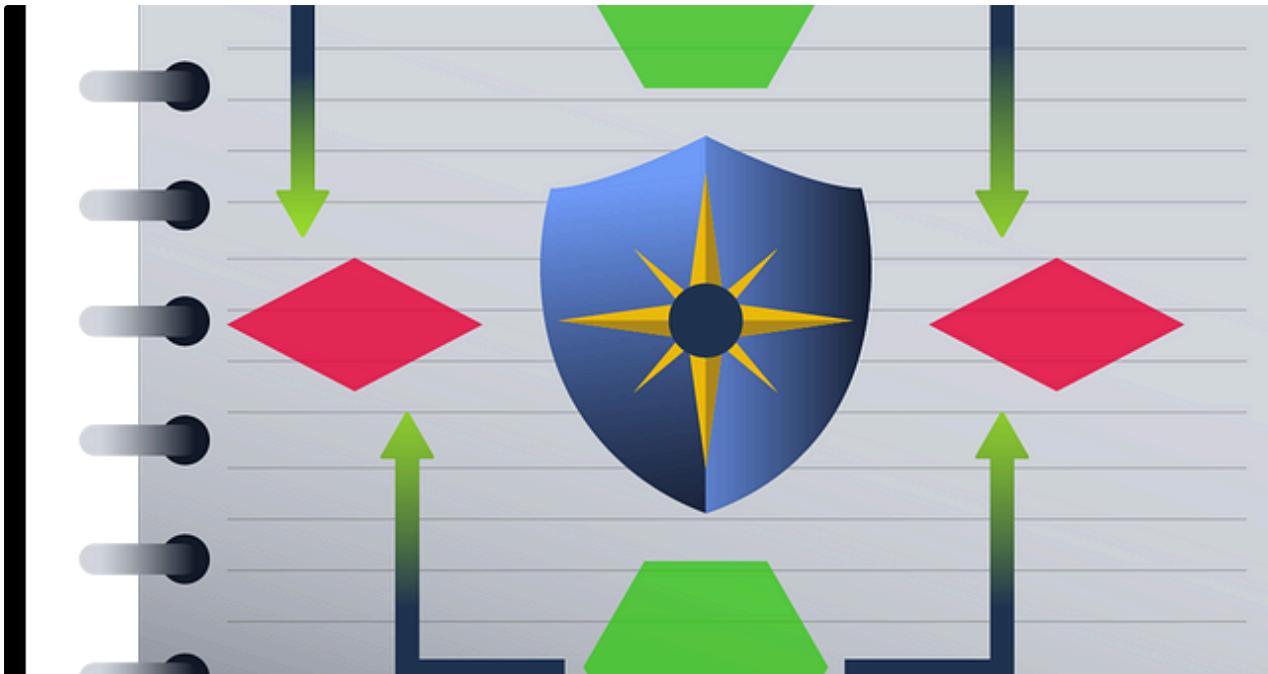
Staff picks

800 stories · 1569 saves



Natural Language Processing

1889 stories · 1546 saves



Sunny Singh Verma [SuNnY]

IR Playbooks TryHackMe Walkthrough Writeup THM |—SuNnY

Kudos to The Creators of this Room :

Sep 13, 2024 100 1



...

Learn > Dead End?

Dead End?

You're given a memory image and a disk image - help us find the flag!

Hard 60 min

Start AttackBox Help Save Room

27



Dead End?— TryHackMe Write-Up

A Step-by-Step Guide to Analyzing Memory and Disk Images in the 'Dead End?' TryHackMe Challenge

Sep 6, 2024 74



APIWizards Breach

Security breach at APIWizards Inc.

90 min

Help

Save Room

50

Options

Room completed (100%)

Board Write-ups

Aakash Raman

TryHackMe APIWizards Breach Walkthrough

This is an interesting room for all the DFIR Enthusiasts on Linux Forensics & Linux Persistence Techniques! Let's get started!

Aug 5, 2024 58



```

HASAN2.E01 - Notepad
File Edit Format View Help
Cylinders: 16,383
Heads: 16
Sectors per Track: 63
Bytes per Sector: 512
Sector Count: 127,800,448
[Physical Drive Information]
Drive Interface Type: ide
[Image]
Image Type: VMWare Virtual Disk
Source data size: 62402 MB
Sector count: 127800448
[Computed Hashes]
MD5 checksum: 3f08c518adb3b5c1359849657a9b2079
SHA1 checksum: d5ae22ab381cb5884140ef6bfab3946a8f3cf9f2

Image Information:
Acquisition started: Mon Feb 8 04:40:23 2021
Acquisition finished: Mon Feb 8 04:47:01 2021
Segment list:
Z:\HASAN2.E01

Image Verification Results:
Verification started: Mon Feb 8 04:47:01 2021
Verification finished: Mon Feb 8 05:06:34 2021
MD5 checksum: 3f08c518adb3b5c1359849657a9b2079 : verified
SHA1 checksum: d5ae22ab381cb5884140ef6bfab3946a8f3cf9f2 : verified

```

Chicken0248

[TryHackMe Write-up] Disk Analysis & Autopsy

Ready for a challenge? Use Autopsy to investigate artifacts from a disk image.

Sep 14, 2024



...

See more recommendations