Investigating with Splunk Workshop

May 2023 | Version 2.2

Based on BOTS 1.0 Dataset

Randy Holloway, CISSP, MS / Staff Sales Engineer

splunk> turn data into doing™

#whoami

Randy Holloway

rholloway@splunk.com

Based in Houston, TX

23+ Years IT and Security Experience

14+ Years SIEM Experience

Came Over from ArcSight

Enjoys Michigan Football and Baseball

Why Investigating?

Scenario Review

Investigating an APT

~~Investigating Ransomware~~

Investigate
(Thanks, Google!)

in·ves·ti·gate

/inˈvestəˌgāt/

verb

verb: **investigate**; 3rd person present: **investigates**; past tense: **investigated**; past participle: **investigated**; gerund or present participle: **investigating**

carry out a systematic or formal inquiry to discover and examine the facts of (an incident, allegation, etc.) so as to establish the truth.
"police are investigating the alleged beating"
synonyms: inquire into, look into, go into, probe, explore, scrutinize, conduct an investigation into, make inquiries about; More

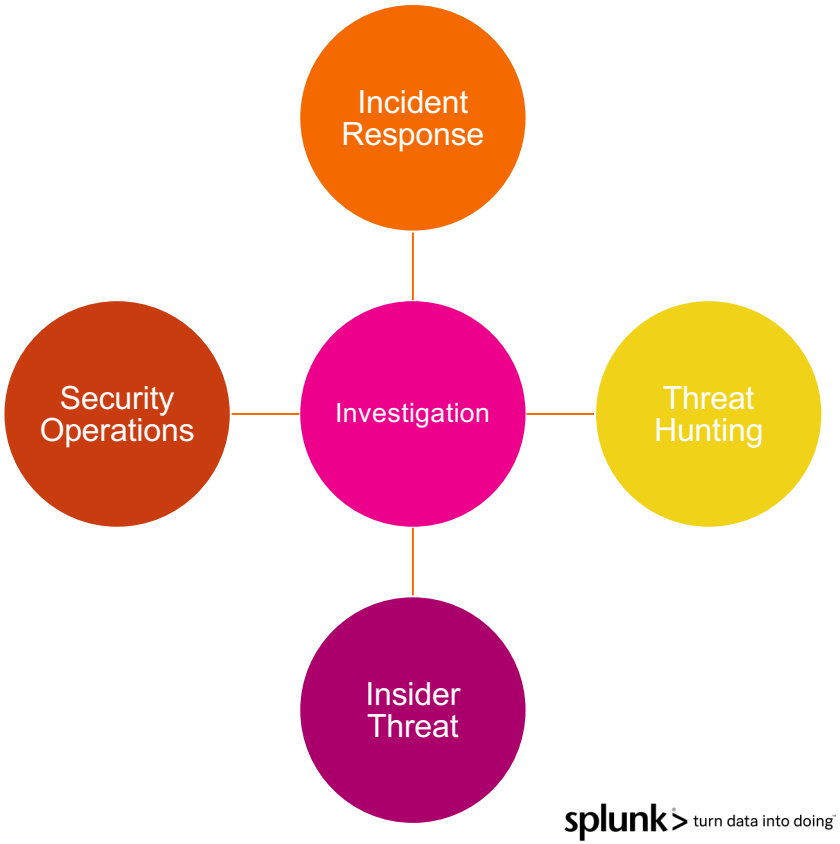
- carry out research or study into (a subject, typically one in a scientific or academic field) so as to discover facts or information.
  "future studies will investigate whether long-term use of the drugs could prevent cancer"
  synonyms: inquire into, look into, go into, probe, explore, scrutinize, conduct an investigation into, make inquiries about; More
- make inquiries as to the character, activities, or background of (someone).
  "everyone with a possible interest in your brother's death must be thoroughly investigated"
  synonyms: inquire into, look into, go into, probe, explore, scrutinize, conduct an investigation into, make inquiries about; More
- make a check to find out something.
  "when you didn't turn up, I thought I'd better come back to investigate"

Investigating Is A Core Competency

When something is detected, we want to figure out what happened

Who, what, where, when, why, how?

Investigating can be reactive or proactive



Using Splunk for Security

Investigations

Luckily Everything Is Captured In Splunk

Microsoft Sysmon

Windows Events

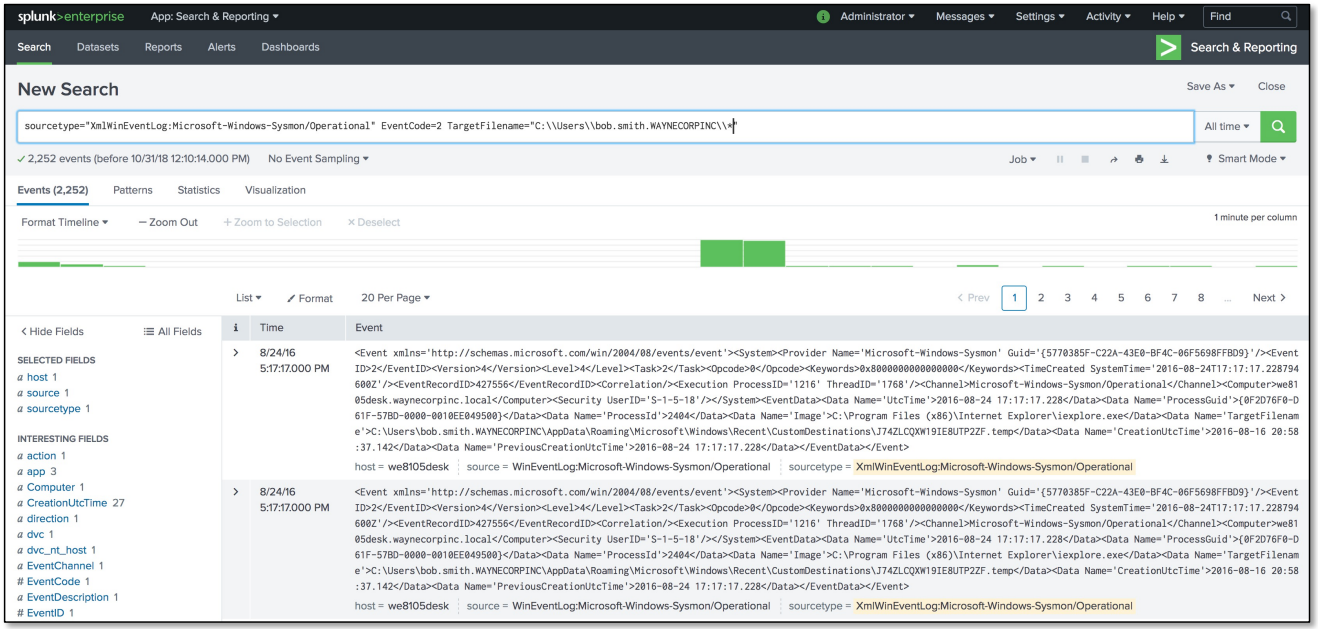
Windows Registry

IIS

Splunk Stream (wire data)

Suricata

Fortigate (NGFW)



Splunk Commands Used

metadata

stats – distinct count, count, values, average, AS

eval – lower, length, round

table

AND OR NOT

sort

reverse

head

transaction

rex

search

inputlookup

outputlookup

fields

lookup

Let's Investigate

splunk> turn data into doing™

Access Class Material

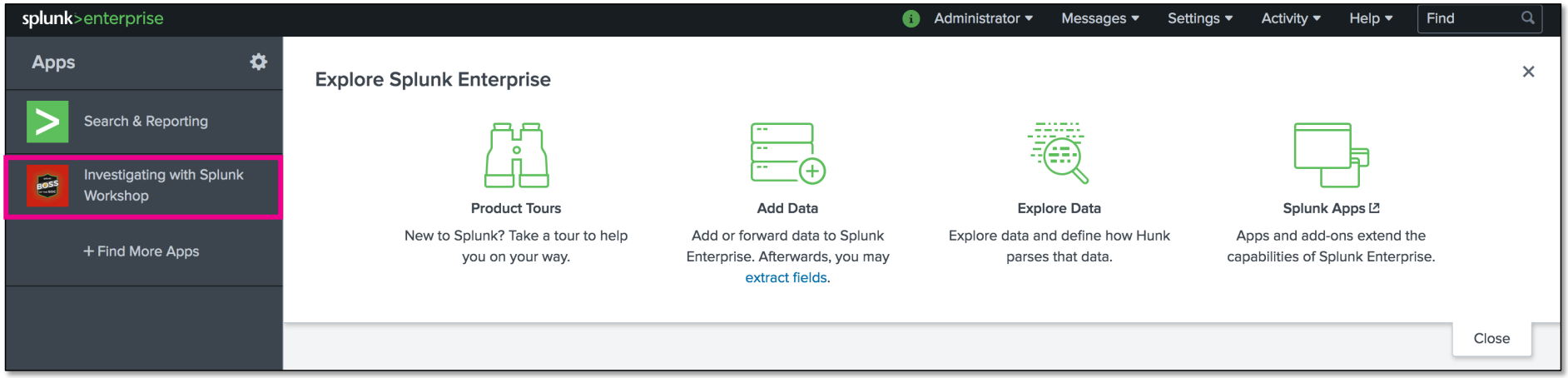
Items of Interest can be found by going to this Google Drive:

1. **This link will have your Lab Guide, Splunk Instance Details and more:**
   https://tinyurl.com/splunkworkshops
2. **Follow the guidance of your instructor on accessing / noting which instance you will use for this workshop, along with getting access to the slides and lab guide.**

Starting Point

Please make sure you are logged in and seeing this window

Use the search & reporting app to hunt. ONLY use the Investigating app AFTER you find your results and you want to verify what you found is correct.



Quick Intro To The App

splunk>enterprise App: Investigating with Splunk Workshop ▾ | Administrator ▾ | Messages ▾ | Settings ▾ | Activity ▾ | Help ▾ | Find

Overview ▾ | Scenario #1 - APT ▾ | Scenario #2 - Ransomware ▾ | Supplemental Material ▾ | Search | Dashboards | Investigating with Splunk Workshop

Introduction

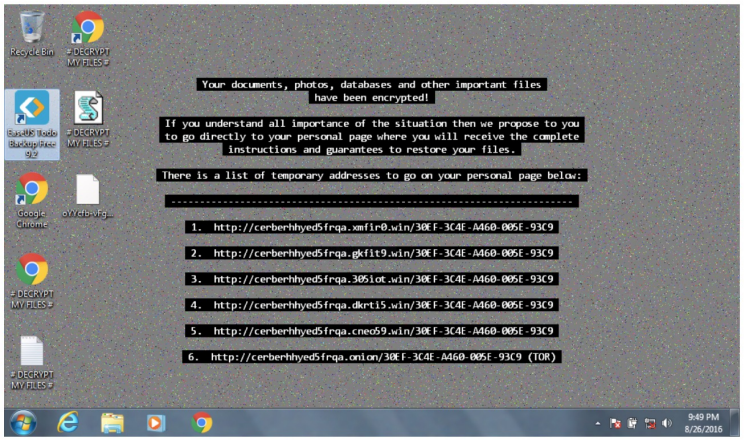
Edit | Export ▾ | ...

Investigating with Splunk Workshop

Welcome to the Investigating with Splunk Workshop based on the Boss of the SOC 2016 data set.

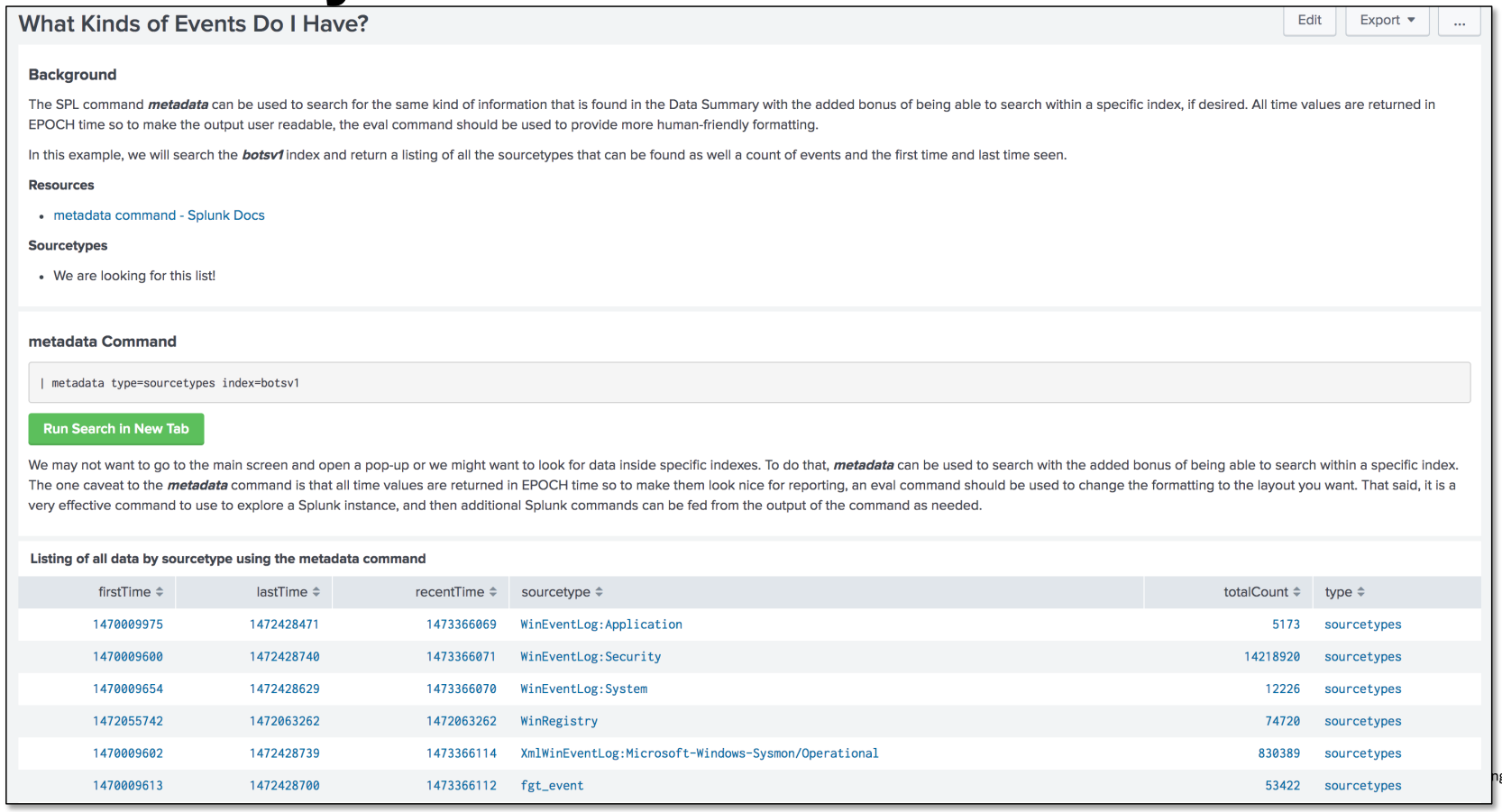
This workshop is designed to provide a hands-on walk through using Splunk as an investigative tool. The focus of this hands on will be an APT scenario and a ransomware scenario.

The hands-on exercise for this workshop is based on the BOTS data set that was developed in 2016 and used for the first iteration of Boss of the SOC. During this workshop, you assume the persona of Alice Bluebird, the analyst who has recently been hired to protect and defend Wayne Enterprises against various forms of cyberattack.



We first suggest you check out the resources below to learn about the environment. Next, review how the dataset was constructed and what sourcetypes exist and why. Finally, work through the hands-on exercises for both A… Ransomware to learn how an adversary would follow a Kill Chain to attack Wayne Enterprises.

Screen Layout



What Kinds of Events Do I Have?

Edit | Export | ...

Background

The SPL command ***metadata*** can be used to search for the same kind of information that is found in the Data Summary with the added bonus of being able to search within a specific index, if desired. All time values are returned in EPOCH time so to make the output user readable, the eval command should be used to provide more human-friendly formatting.

In this example, we will search the ***botsv1*** index and return a listing of all the sourcetypes that can be found as well a count of events and the first time and last time seen.

Resources

- metadata command - Splunk Docs

Sourcetypes

- We are looking for this list!

metadata Command

```
| metadata type=sourcetypes index=botsv1
```

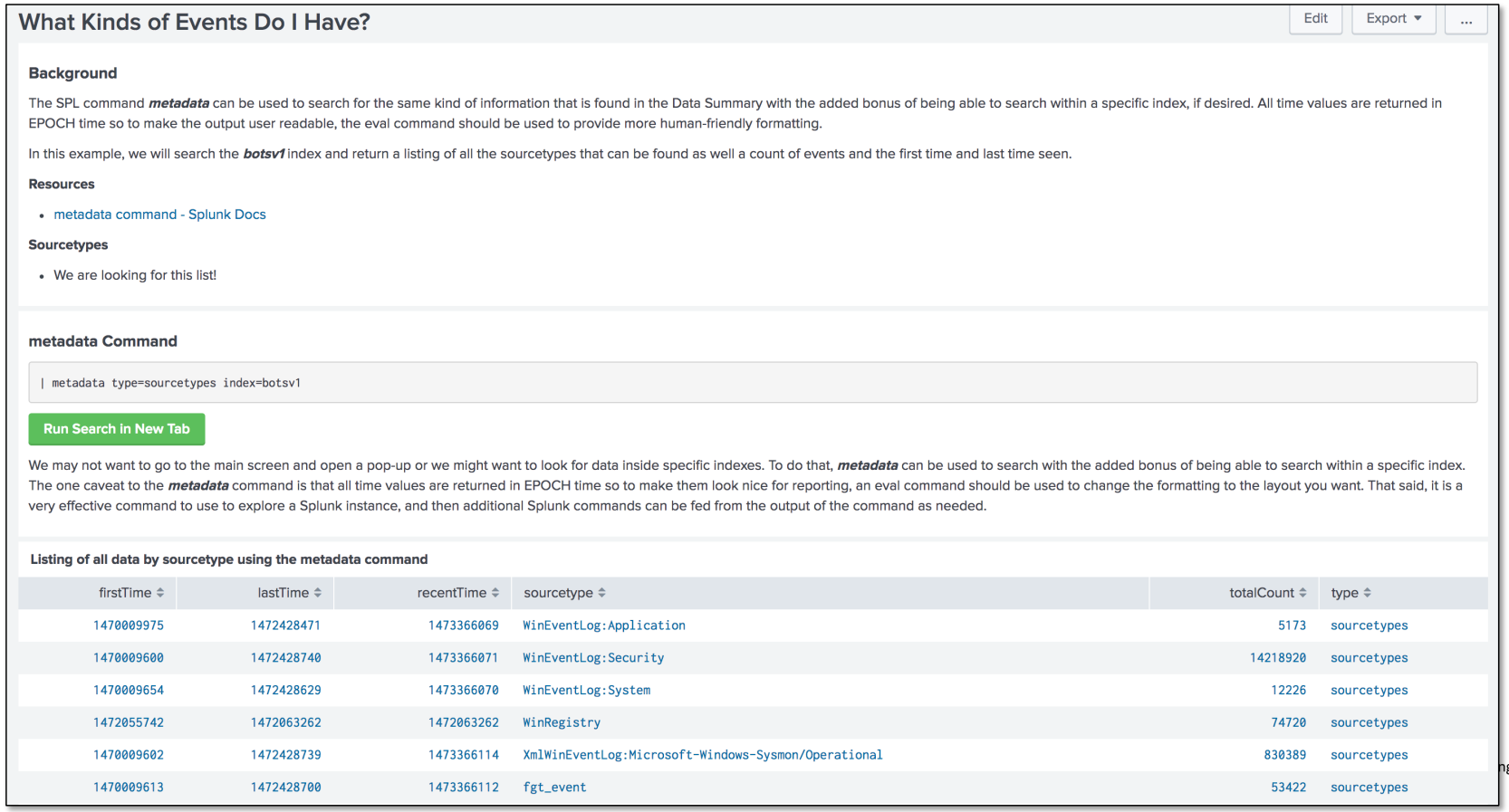
Run Search in New Tab

We may not want to go to the main screen and open a pop-up or we might want to look for data inside specific indexes. To do that, ***metadata*** can be used to search with the added bonus of being able to search within a specific index. The one caveat to the ***metadata*** command is that all time values are returned in EPOCH time so to make them look nice for reporting, an eval command should be used to change the formatting to the layout you want. That said, it is a very effective command to use to explore a Splunk instance, and then additional Splunk commands can be fed from the output of the command as needed.

Listing of all data by sourcetype using the metadata command

firstTime	lastTime	recentTime	sourcetype	totalCount	type
1470009975	1472428471	1473366069	WinEventLog:Application	5173	sourcetypes
1470009600	1472428740	1473366071	WinEventLog:Security	14218920	sourcetypes
1470009654	1472428629	1473366070	WinEventLog:System	12226	sourcetypes
1472055742	1472063262	1472063262	WinRegistry	74720	sourcetypes
1470009602	1472428739	1473366114	XmlWinEventLog:Microsoft-Windows-Sysmon/Operational	830389	sourcetypes
1470009613	1472428700	1473366112	fgt_event	53422	sourcetypes

What Kinds of Events Do I Have?



What Kinds of Events Do I Have?

Edit | Export | ...

Background

The SPL command ***metadata*** can be used to search for the same kind of information that is found in the Data Summary with the added bonus of being able to search within a specific index, if desired. All time values are returned in EPOCH time so to make the output user readable, the eval command should be used to provide more human-friendly formatting.

In this example, we will search the ***botsv1*** index and return a listing of all the sourcetypes that can be found as well a count of events and the first time and last time seen.

Resources

- metadata command - Splunk Docs

Sourcetypes

- We are looking for this list!

metadata Command

```
| metadata type=sourcetypes index=botsv1
```

Run Search in New Tab

We may not want to go to the main screen and open a pop-up or we might want to look for data inside specific indexes. To do that, ***metadata*** can be used to search with the added bonus of being able to search within a specific index. The one caveat to the ***metadata*** command is that all time values are returned in EPOCH time so to make them look nice for reporting, an eval command should be used to change the formatting to the layout you want. That said, it is a very effective command to use to explore a Splunk instance, and then additional Splunk commands can be fed from the output of the command as needed.

Listing of all data by sourcetype using the metadata command

firstTime	lastTime	recentTime	sourcetype	totalCount	type
1470009975	1472428471	1473366069	WinEventLog:Application	5173	sourcetypes
1470009600	1472428740	1473366071	WinEventLog:Security	14218920	sourcetypes
1470009654	1472428629	1473366070	WinEventLog:System	12226	sourcetypes
1472055742	1472063262	1472063262	WinRegistry	74720	sourcetypes
1470009602	1472428739	1473366114	XmlWinEventLog:Microsoft-Windows-Sysmon/Operational	830389	sourcetypes
1470009613	1472428700	1473366112	fgt_event	53422	sourcetypes

APT Scenario

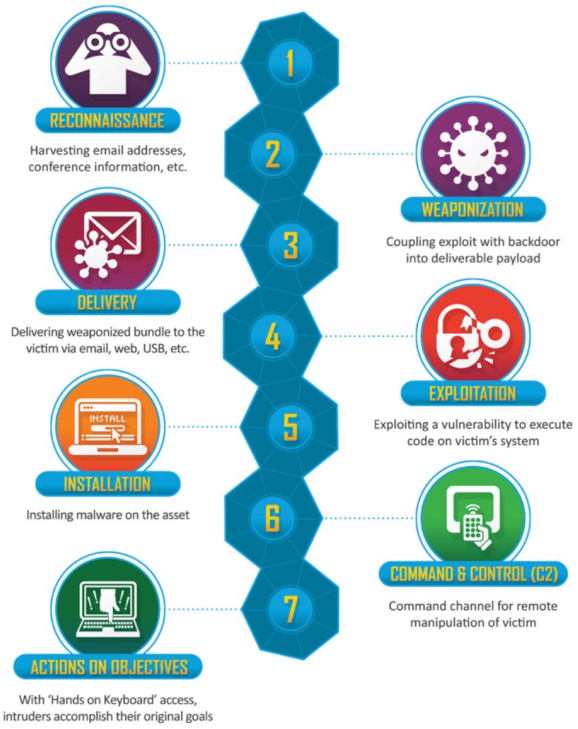
splunk> turn data into doing™

APT Prologue

Edit | Export | ...

Scenario #1 - APT

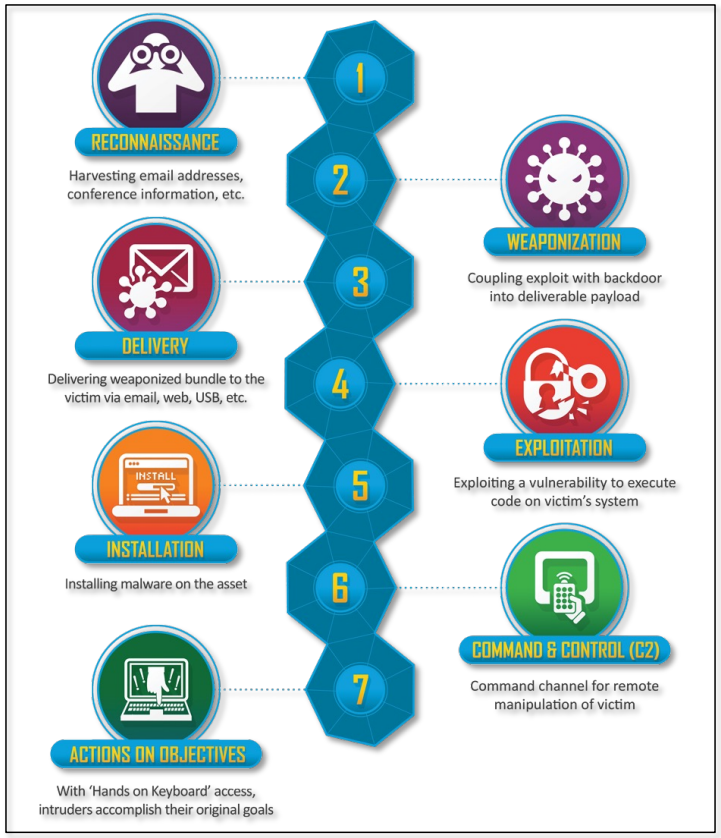
In this scenario, reports of the below graphic come in from your user community when they visit the Wayne Enterprises website, and some of the reports reference "P01s0n1vy." In case you are unaware, P01s0n1vy is an APT group that has targeted Wayne Enterprises. Your goal, as Alice, is to investigate the defacement, with an eye towards reconstructing the attack via the Lockheed Martin Kill Chain.



Lockheed Martin Cyber Kill Chain

Method for characterizing different parts of an attack for intelligence gathering, situational awareness and better understanding of a multi-stage threat

Does not apply to all threats but is a good place to start for targeted attacks



http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf

Start Investigating!

During an investigation, there are a number of questions that need to be answered.

- Some questions will build off of earlier questions
- Pro tip: have a notepad and pencil at the ready!

We created these questions to guide your investigation, there are more questions you can ask and more badness you can find!

- After each question, the instructor will set aside a few minutes to allow you to find the answer
- We will re-group to review the question and the process we took
- Reference the app if you get stuck or ask your instructors
- We will provide a couple of hints to get you started...

Finding the IP Scanning Your Web Server

Kill Chain Phase: Reconnaissance

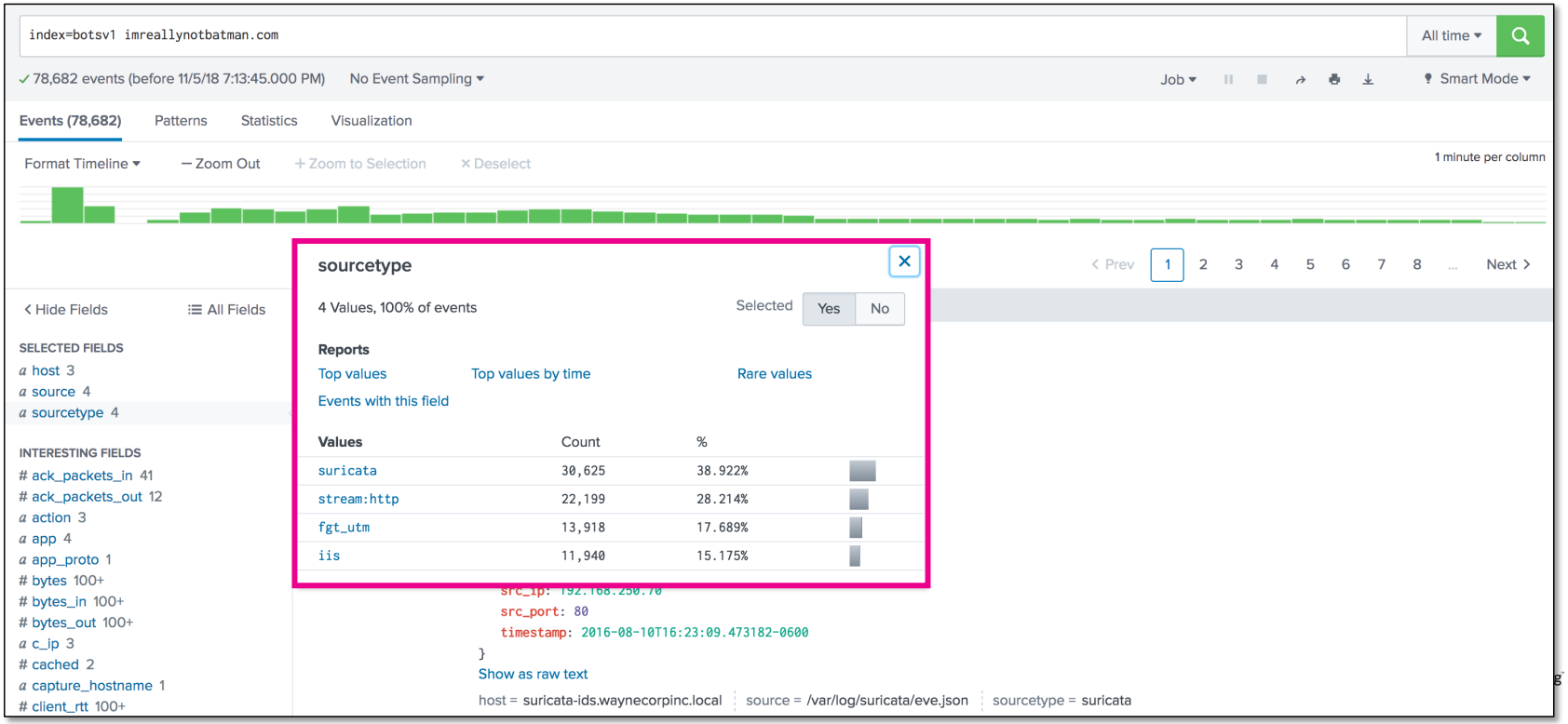
What is the likely IP address of someone from the Po1s0n1vy group scanning imreallynotbatman.com for web application vulnerabilities?

Hints

- All searches used today will have an index of botsv1
  - index=botsv1
- The field src is the field you are looking for
- Focusing your search on specific sourcetypes will be helpful

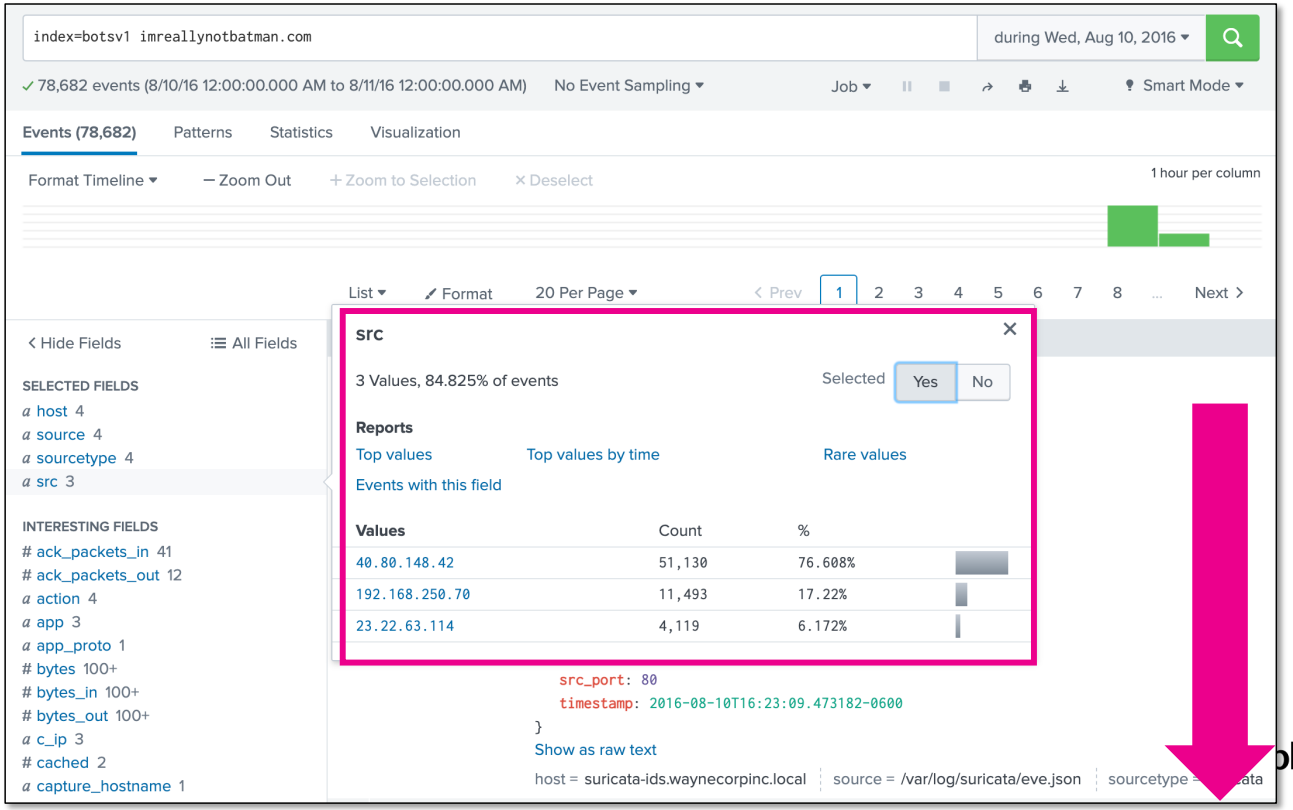
Identify sourcetypes Associated with Search Values

Kill Chain Phase: Reconnaissance



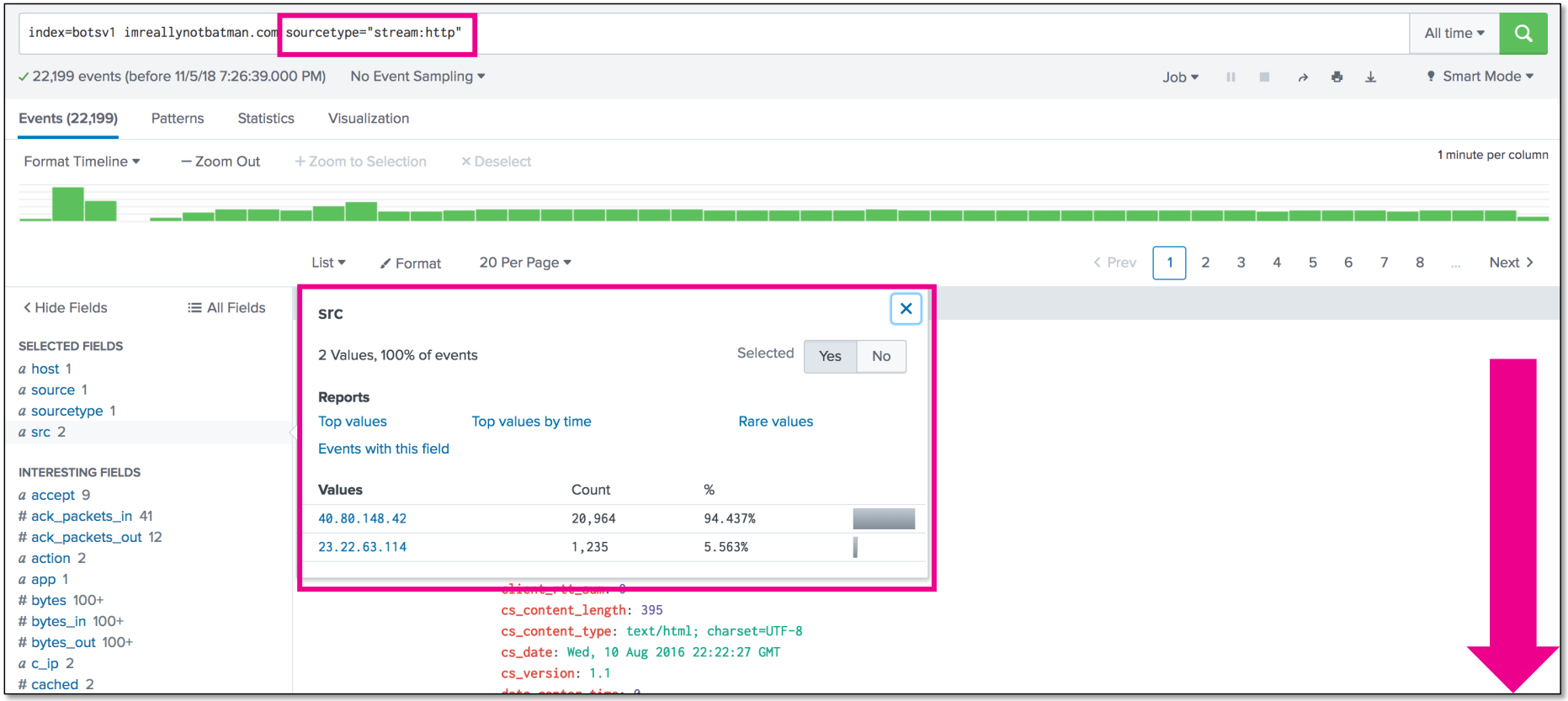
Finding Source Addresses

Kill Chain Phase: Reconnaissance

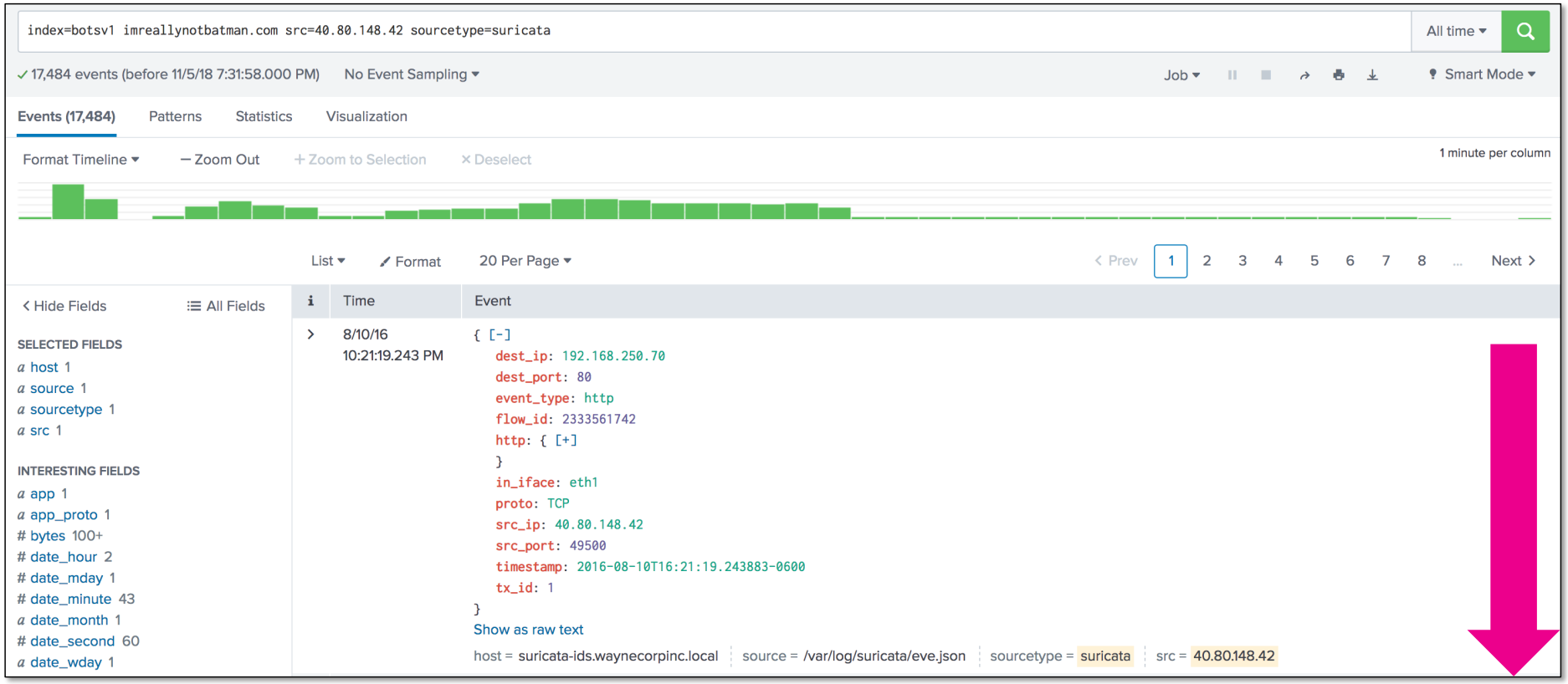


splunk> turn data into doing

Selecting a sourcetype and Searching for Source Address



Validation with suricata

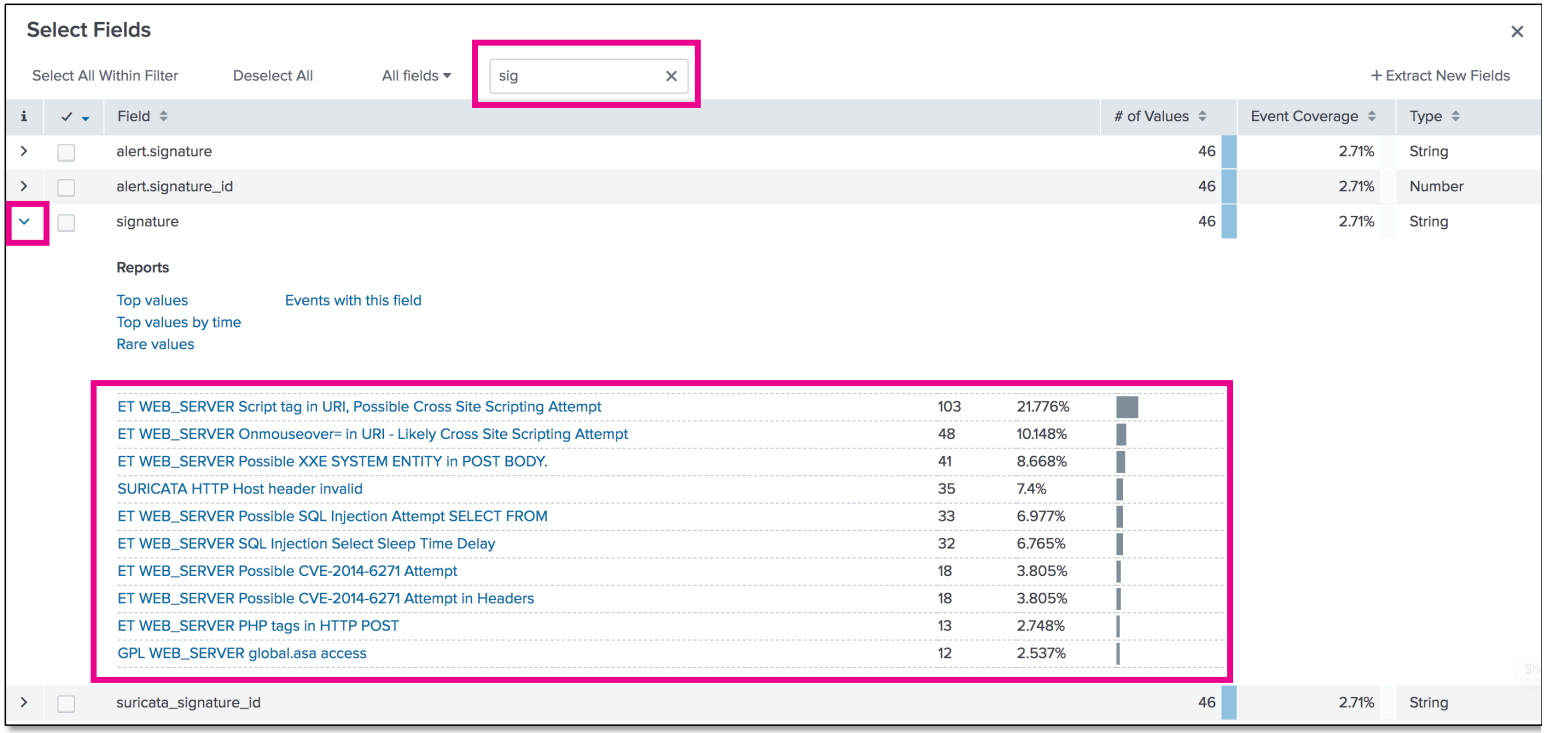


More Fields...

a tag 3
a tag::eventtype 3
timeendpos 1
a timestamp 100+
timestartpos 1
a transport 1
a url 100+
a vendor 1

19 more fields

+ Extract New Fields



Finding the IP Scanning Your Web Server

Kill Chain Phase: Reconnaissance

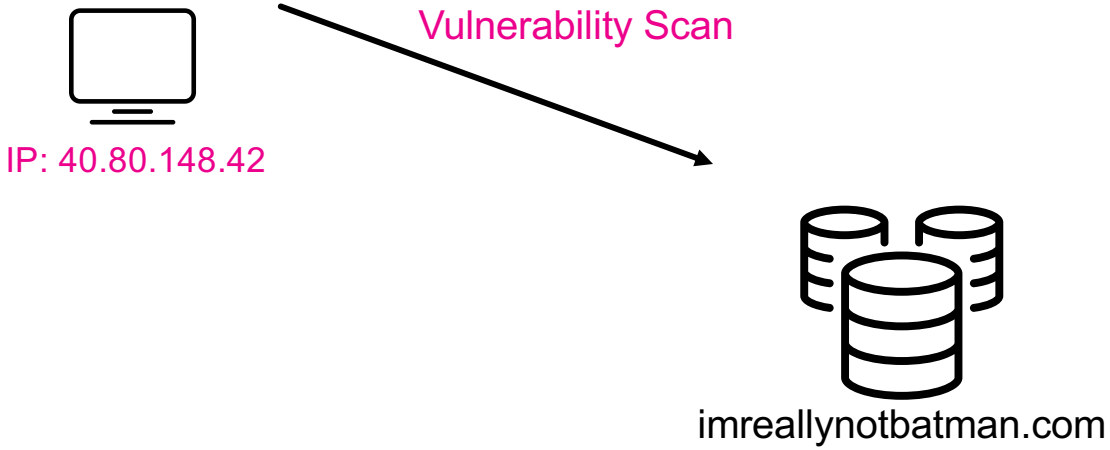
What is the likely IP address of someone from the Po1s0n1vy group scanning imreallynotbatman.com for web application vulnerabilities?

- 40.80.148.42

Kill Chain

IP Scanning the Web Server: 40.80.148.42

APT Picture



Identifying The Web Vulnerability Scanner

Kill Chain Phase: Reconnaissance

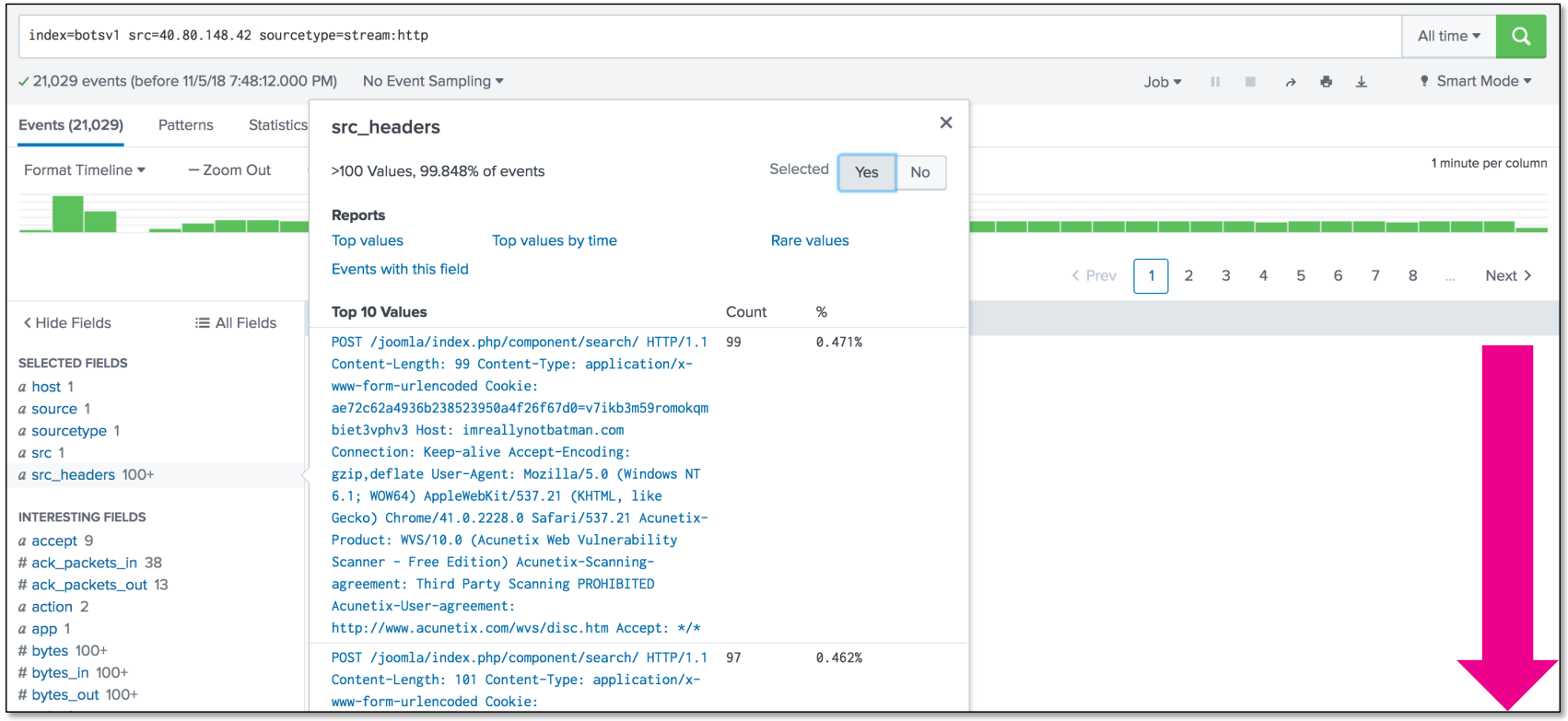
What company created the web vulnerability scanner used by Po1s0n1vy?

Hints

- Use the src found in the prior question as the source of the scans
- The sourcetype stream:http provides a good deal of insight into communications between a web browser and server
- User agent strings can provide value here
- If you don’t know what a term is, Google it!

Looking at src_headers

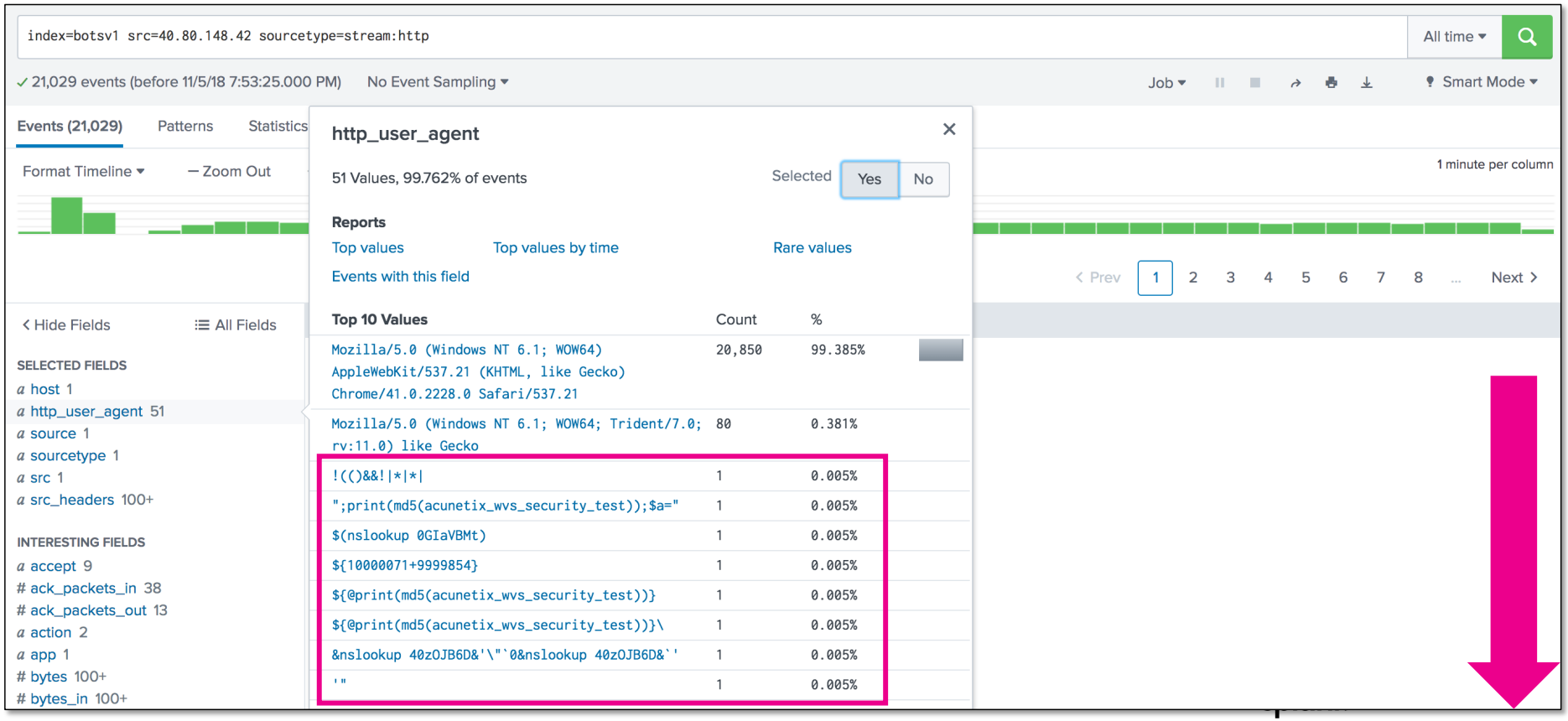
Kill Chain Phase: Reconnaissance



Looking at src_headers

Top 10 Values	Count	%
POST /joomla/index.php/component/search/ HTTP/1.1 Content-Length: 99 Content-Type: application/x-www-form-urlencoded Cookie: ae72c62a4936b238523950a4f26f67d0=v7ikb3m59romokqmbiet3vphv3 Host: imreallynotbatman.com Connection: Keep-alive Accept-Encoding: gzip,deflate 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: */*	99	0.471%
POST /joomla/index.php/component/search/ HTTP/1.1 Content-Length: 101 Content-Type: application/x-www-form-urlencoded Cookie: ae72c62a4936b238523950a4f26f67d0=v7ikb3m59romokqmbiet3vphv3 Host: imreallynotbatman.com Connection: Keep-alive Accept-Encoding: gzip,deflate 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: */*	97	0.462%

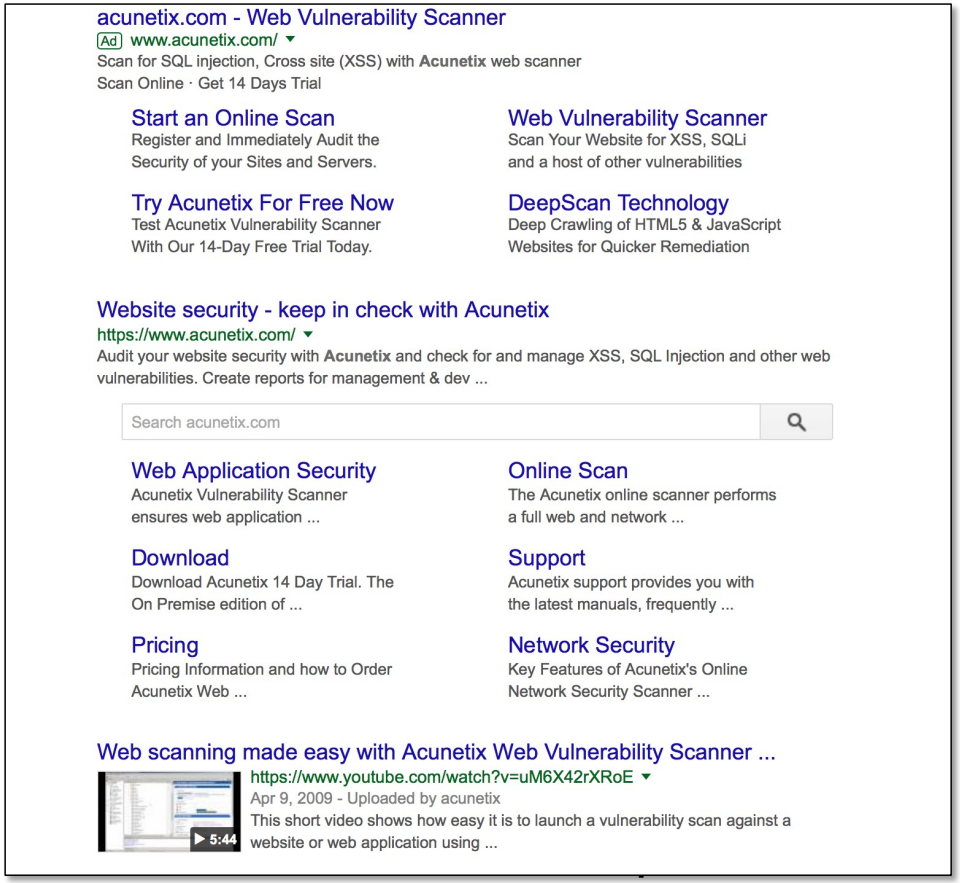
Looking at http_user_agent strings



Research on the Internet

If you see something in the logs you don't understand, Google it!

What If You Don't Know What Acunetix Is?



Identifying The Web Vulnerability Scanner

Kill Chain Phase: Reconnaissance

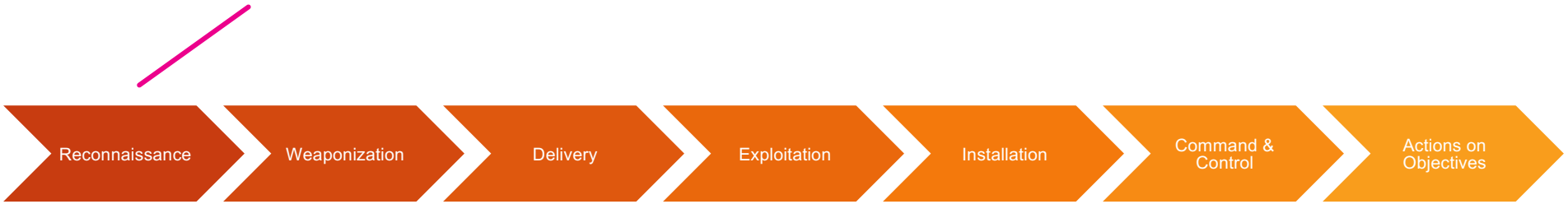
What company created the web vulnerability scanner used by P01s0n1vy?

- Acunetix

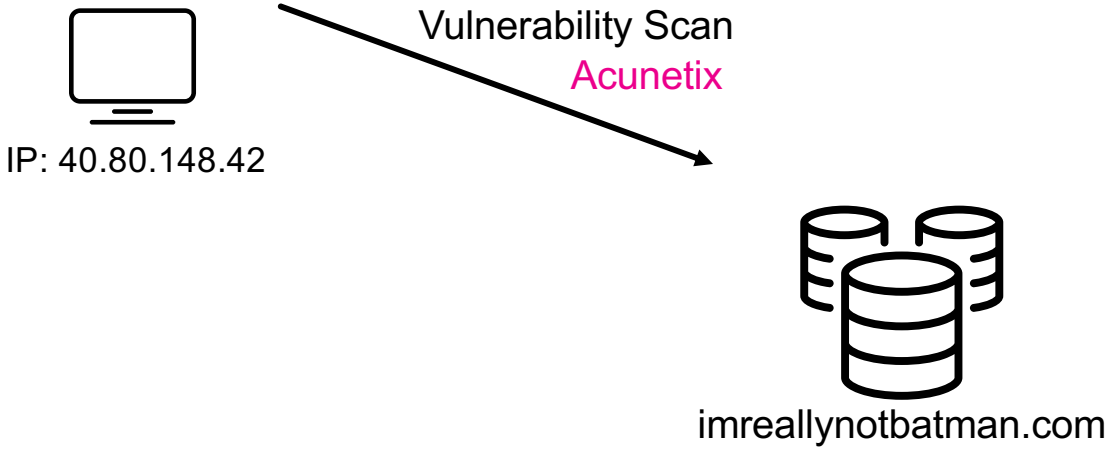
Kill Chain

IP Scanning the Web Server: 40.80.148.42

Web Vulnerability Scanner: Acunetix



APT Picture



Determining Which Web Server is the Target

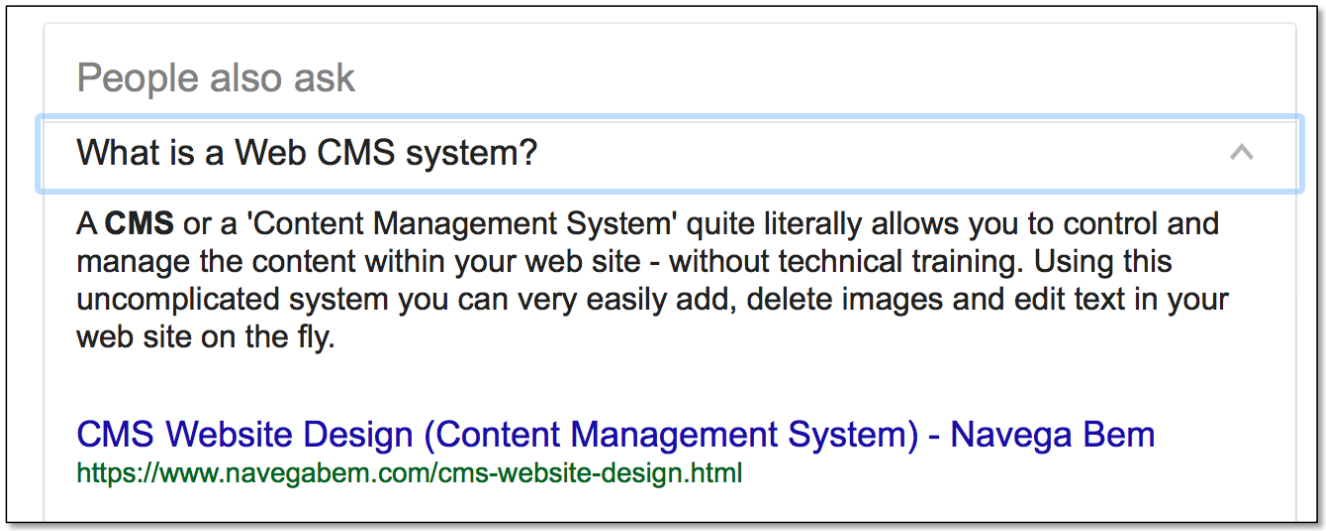
Kill Chain Phase: Reconnaissance

What content management system is imreallynotbatman.com likely using?

Hints

- If you aren't sure what a content management system is, Google it.
- Sourcetypes that are relevant to web communication are helpful
- URIs contain information that will lead you in the right direction

Where to start?

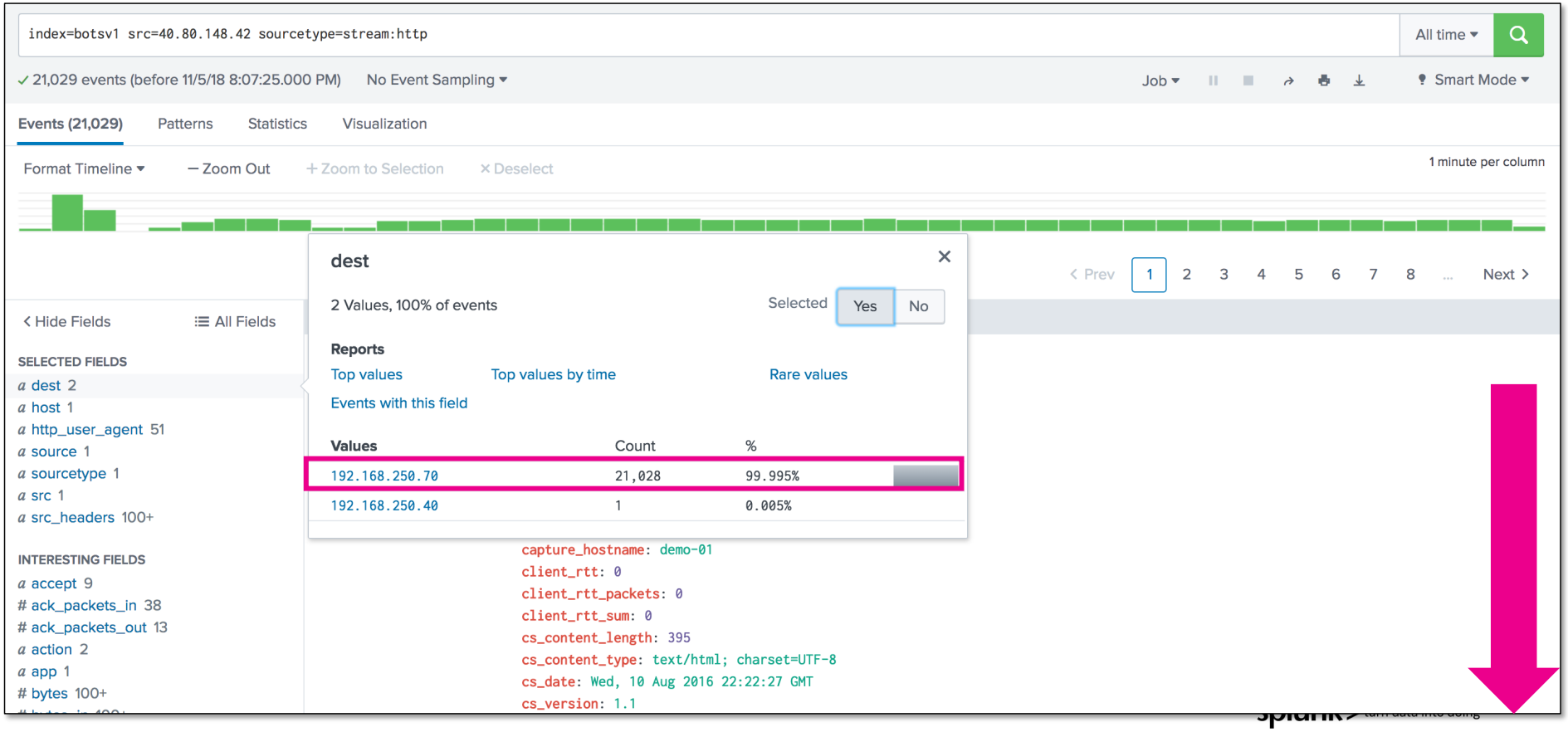


Based on that in what kind of logs should I look?

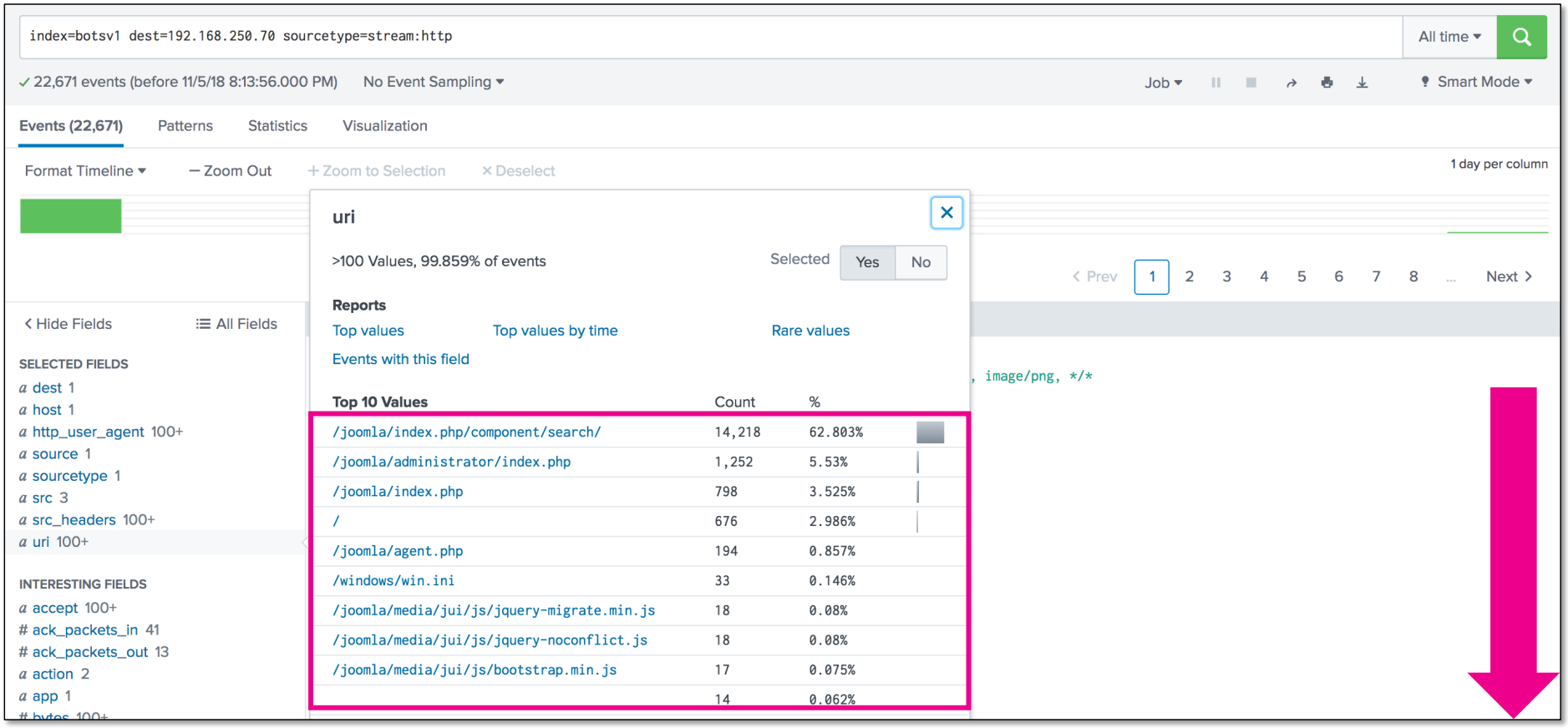
Do we have the IP of the system in question?

Based on our previous findings, probably so...

Where To Start?



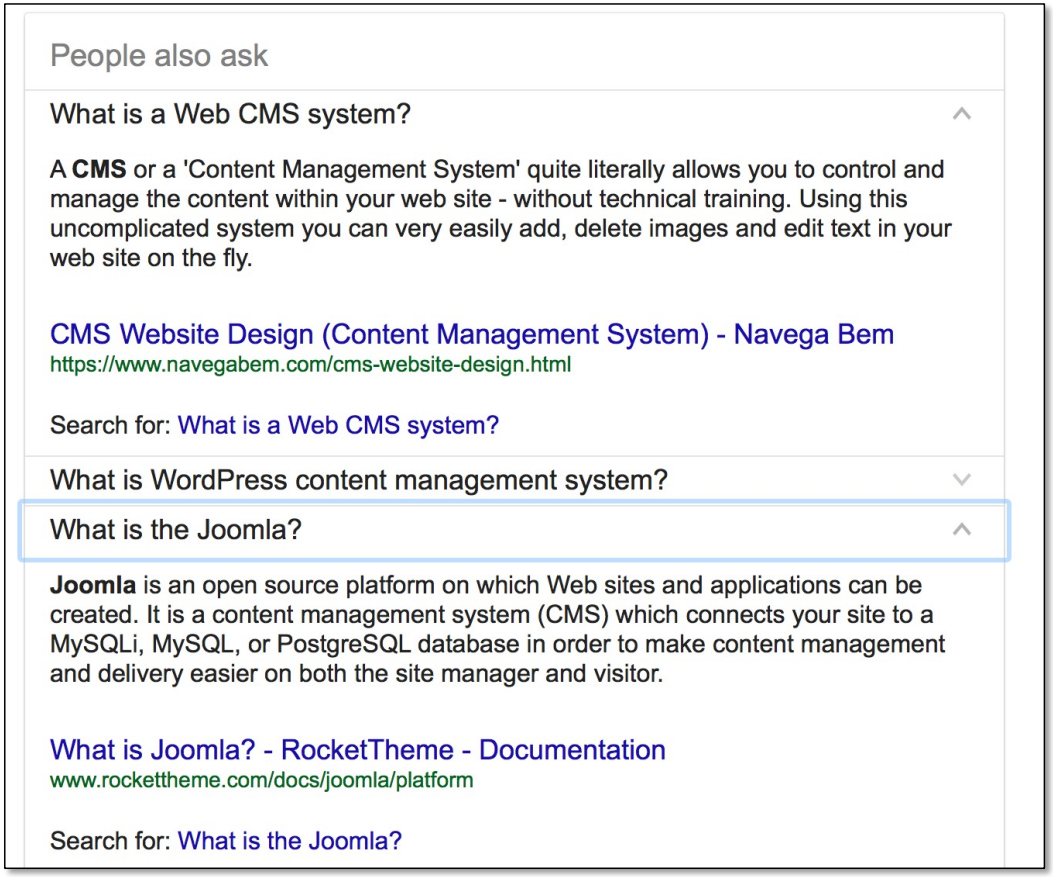
Digging into the URI



Digging into the URI

External Research

Much like the previous question, if you see values you have never seen before, Google is your friend



Looking for Confirmation

Add The Response Code condition to ensure we have successful page loads

```
index=botsv1 dest=192.168.250.70 sourcetype=stream:http status=200 | stats count by uri | sort - count
```

Generate a count by URI to see what values are most frequently seen

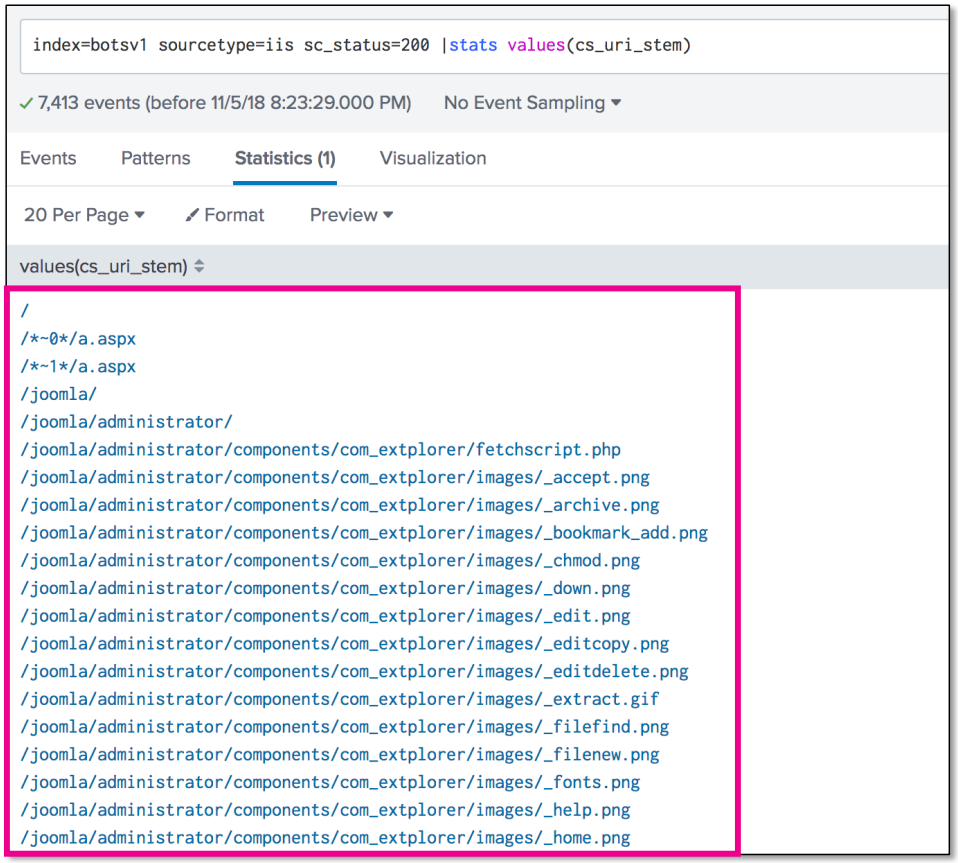
Sort descending by count so the greatest values come to the top

Our Results? Lots of Joomla

```
index=botsv1 dest=192.168.250.70 sourcetype=stream:http status=200 | stats count by uri | sort - count
```

uri	count
/joomla/index.php/component/search/	2207
/joomla/administrator/index.php	840
/joomla/index.php	688
/	610
/joomla/agent.php	193
/windows/win.ini	33
/joomla/media/jui/js/jquery-migrate.min.js	17
/joomla/media/jui/js/jquery-noconflict.js	17
/joomla/media/jui/js/bootstrap.min.js	16
/joomla/media/system/js/html5fallback.js	13

Finding the Answer With IIS



Determining Which Web Server is the Target

Kill Chain Phase: Reconnaissance

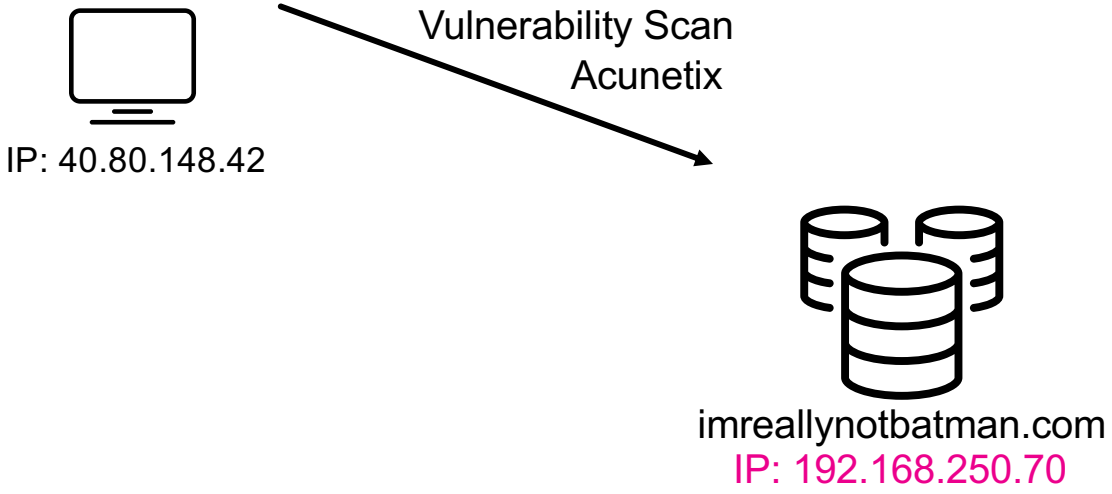
What content management system is imreallynotbatman.com likely using?

- Joomla

Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

APT Picture



Identifying Where a Brute Force Attack Originated

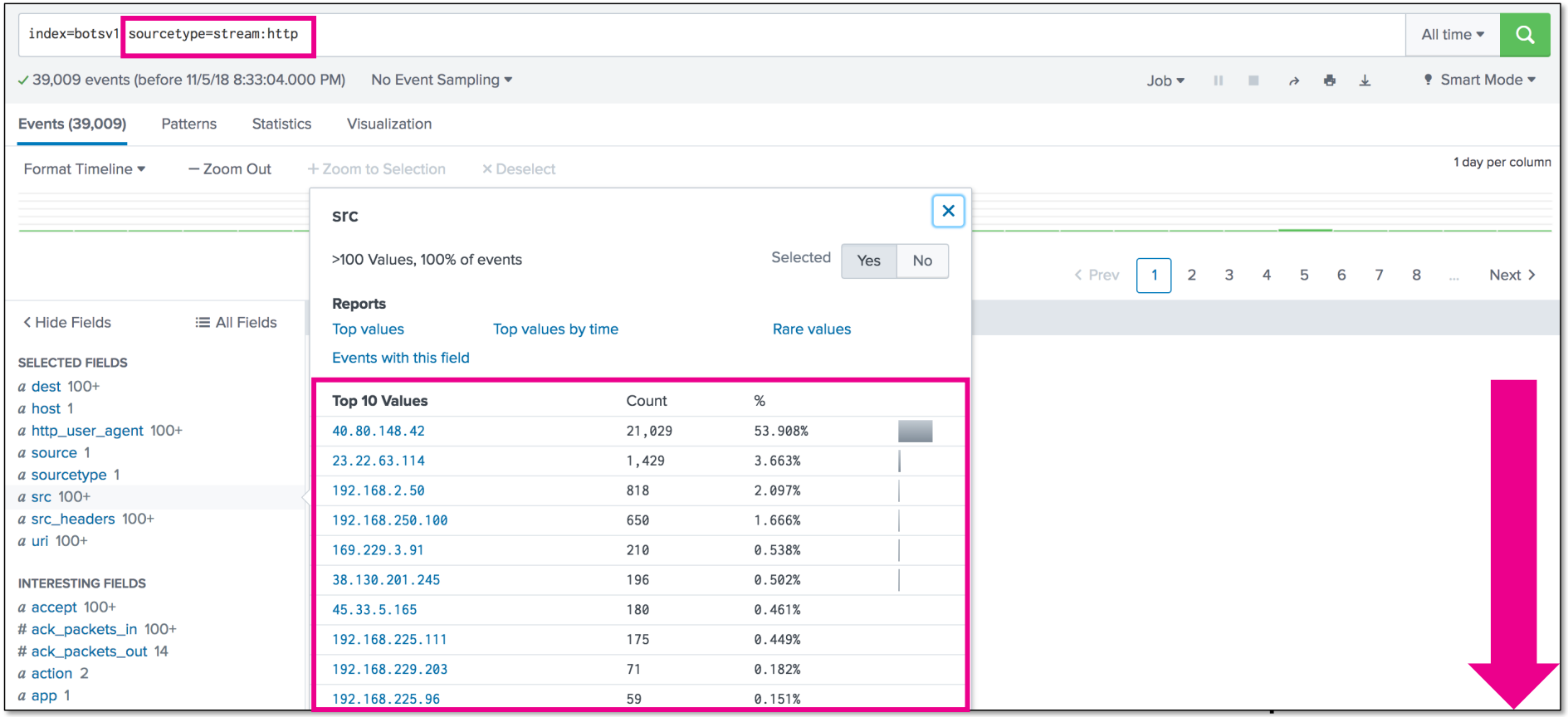
Kill Chain Phase: Exploitation

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

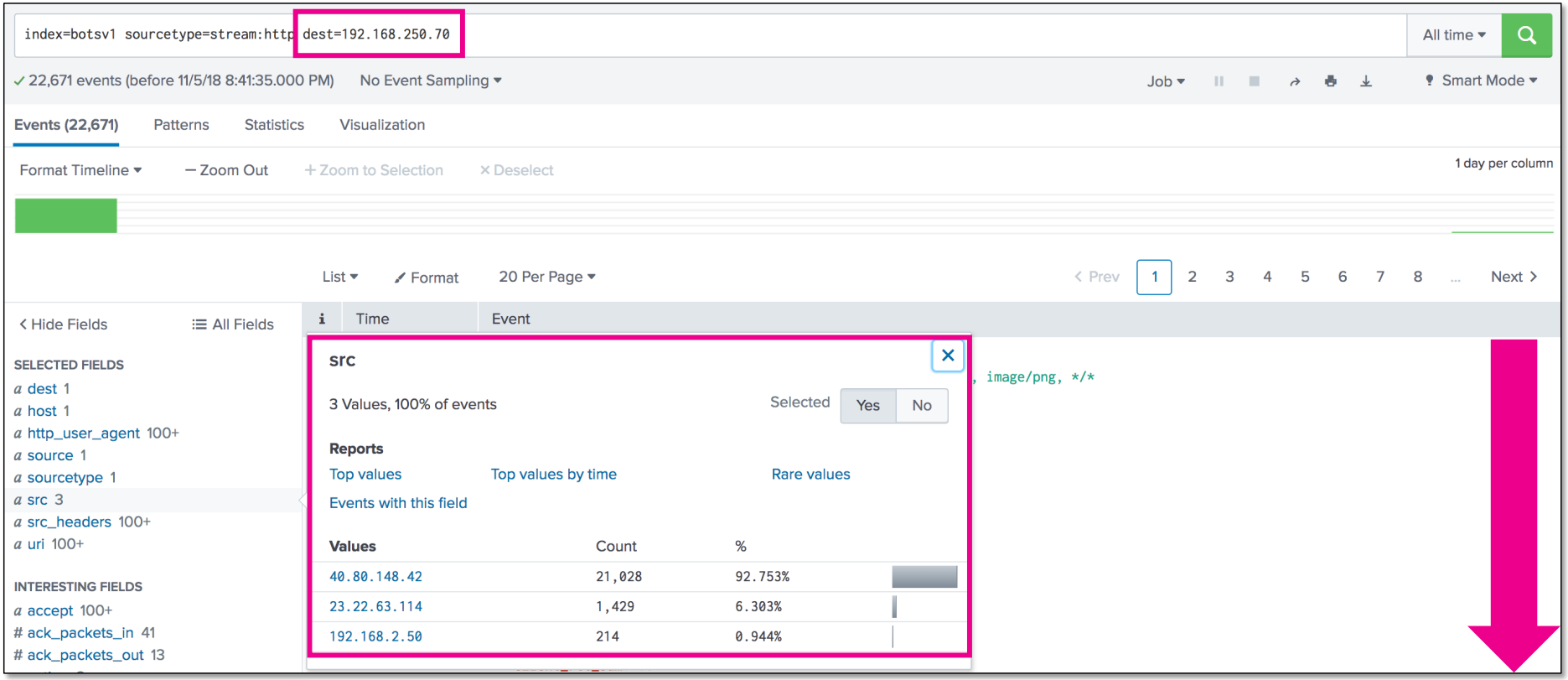
Hints

- A sourcetype that can see web browser to server communication would be very helpful
- Knowing the destination IP address of the web server being attacked would be good
- The web method will be helpful to focus our search
- The field form_data will have user names and passwords in it if you look closely

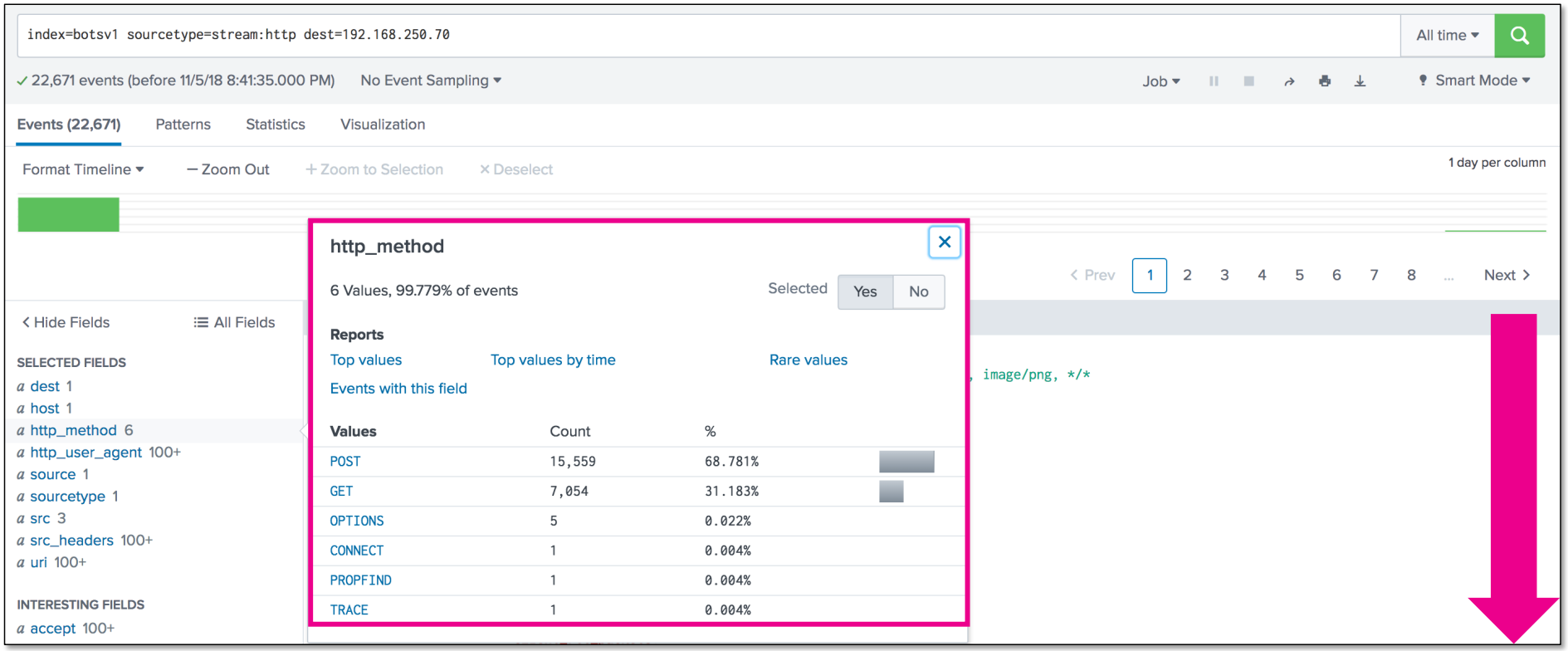
Looking in Wire Data - stream:http



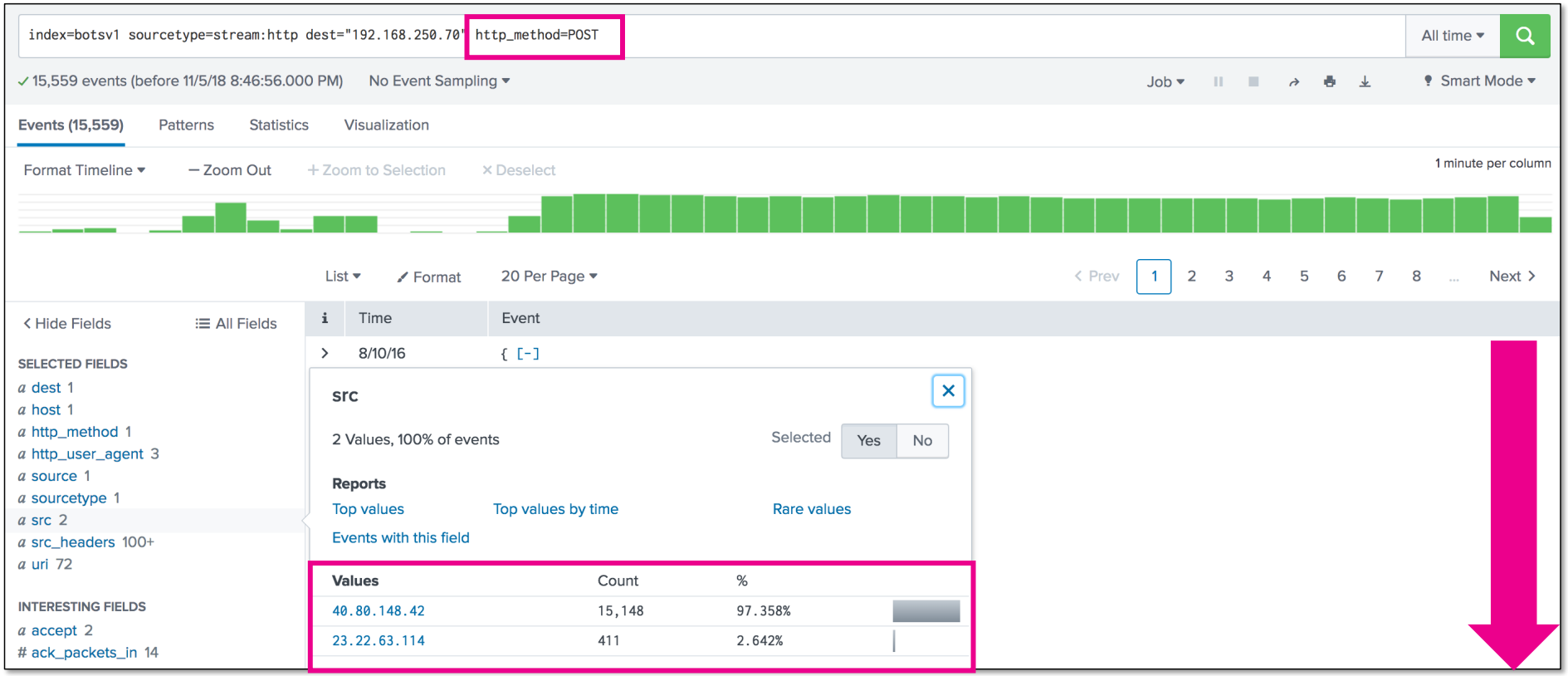
Refine Our Search with the Web Server Address



Same Search, Different Pivot



Adding a HTTP Method to Narrow Our Results



Finding Passwords in HTTP Wire Data

```
index=botsv1 sourcetype=stream:http dest="192.168.250.70" http_method=POST form_data=*username*passwd*
```

dest	192.168.250.70
dest_content	<head><title>Document Moved</title></head> <body> Object Moved This document may be found <a HREF="http://imreallynotbatman.com/joomla/administrator/index.php">here</body>
dest_headers	HTTP/1.1 303 See other Content-Type: text/html; charset=UTF-8 Location: http://imreallynotbatman.com/joomla/administrator/index.php Server: Microsoft-IIS/8.5 X-Powered-By: PHP/5.5.38 Date: Wed, 10 Aug 2016 21:48:05 GMT Content-Length: 182
dest_ip	192.168.250.70
dest_mac	00:0C:29:C4:02:7E
dest_port	80
duplicate_packets_in	1
duplicate_packets_out	1
duration	1754955
endtime	2016-08-10T21:48:05.858372Z
eventtype	stream_network_traffic (communicate network) stream_web (web)
form_data	username=admin&passwd=batman&option=com_login&task=login&return=aW5kZXgucGhw&e5ec827a3f67ce0efc546d81f7356acc=1
http_comment	HTTP/1.1 303 See other

Finding Passwords in HTTP Wire Data

```
index=botsv1 sourcetype=stream:http dest="192.168.250.70" http_method=POST form_data=*username*passwd*
|table form_data
```

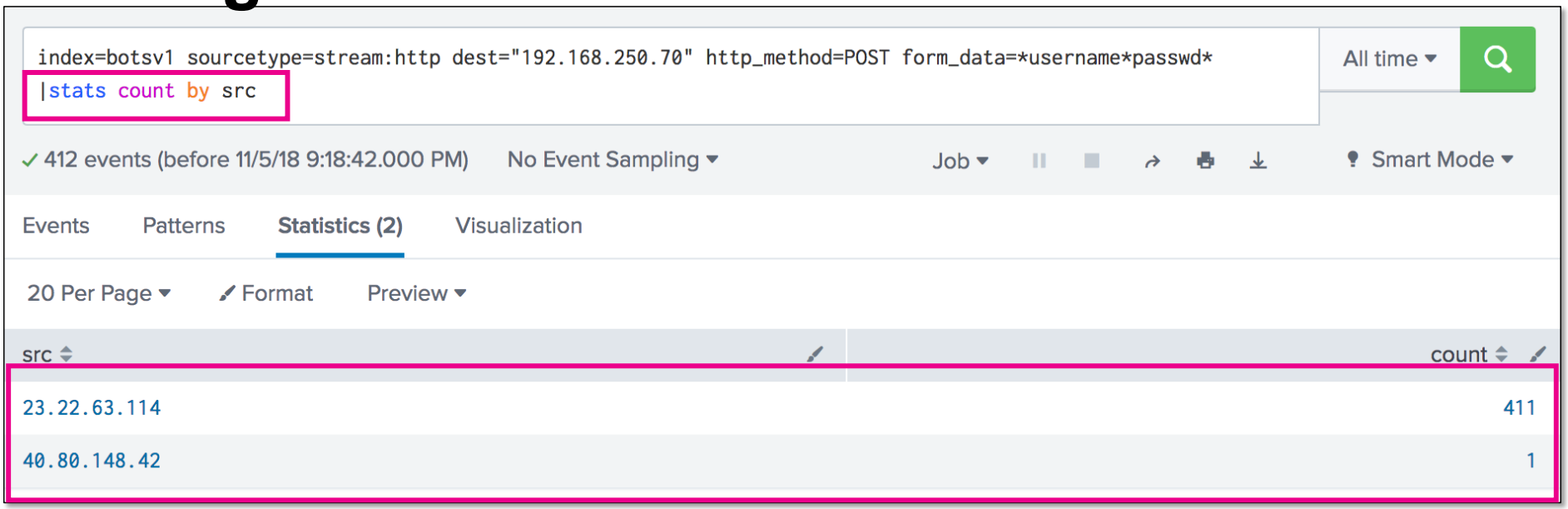
✓ 412 events (before 11/5/18 9:13:58.000 PM) No Event Sampling

Events Patterns **Statistics (412)** Visualization

20 Per Page Format Preview

form_data
username=admin&passwd=batman&option=com_login&task=login&return=aW5kZXgucGhw&e5ec827a3f67ce0efc546d81f7356acc=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=rock&4a40c518220c1993f0e02dc4712c5794=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=cool&a09349d0d6bdbf078ad72cf8e9348583=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=sammy&0d3bb0020f70044ffba32f7d0fa7fa88=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=august&9800c58b682f234e562dee5972a58b8d=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=phantom&a083bf4d12c07976186d8a6efa6308cf=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=williams&e3b1998d29669e83333a101735fd1c90=1
username=admin&ba11501d963f628dfb862d3a07bbe674=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=private
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=baby&26a9247d113c378cdf06f31fa2154f2c=1
username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=dave&1b067a8762b4c8a9909ca68aae723e5a=1

Using stats for the Win!



Identifying Where a Brute Force Attack Originated

Exploitation

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

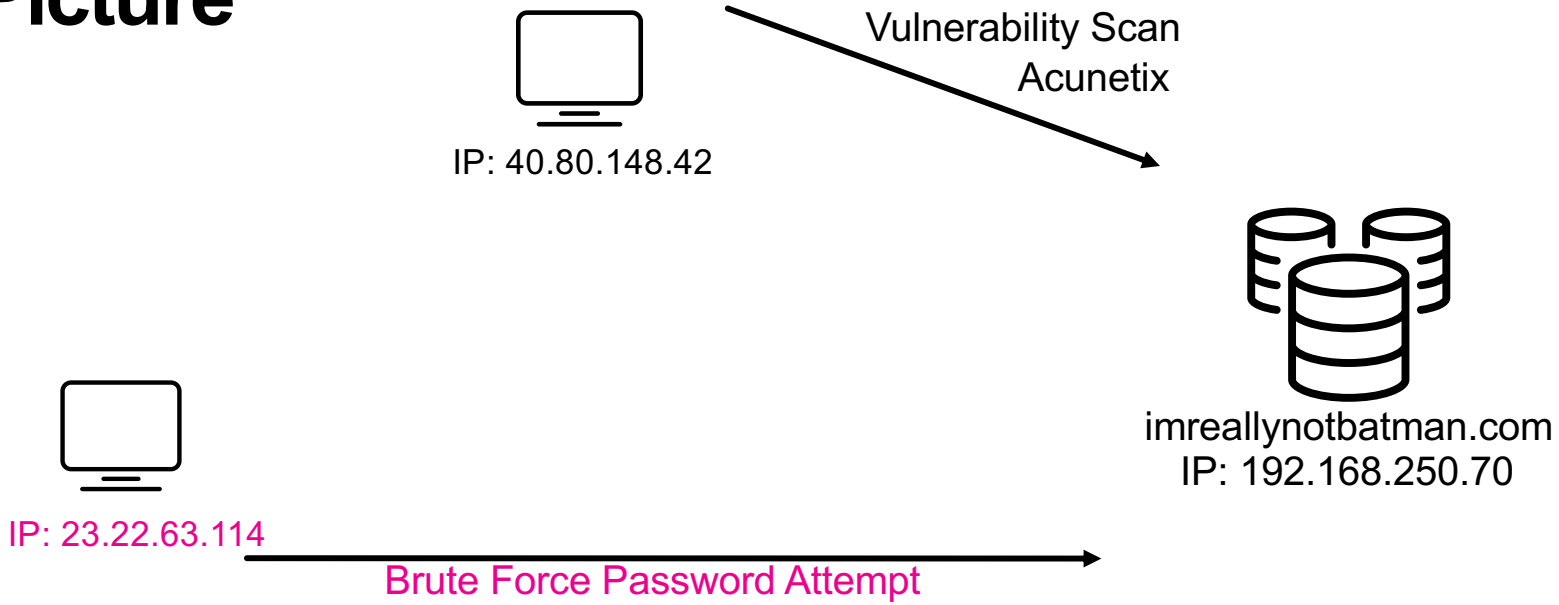
- 23.22.63.114

Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Brute Force Attack Originated: 23.22.63.114

APT Picture



MITRE ATT&CK

Adversarial Tactics, Techniques, and Common Knowledge

Reconnaissance
Resource Development

Initial Access
Execution
Persistence
Privilege Escalation

Defense Evasion
Credential Access
Discovery
Lateral Movement

Collection
Command and Control
Exfiltration
Impact

"The 14 tactic categories within ATT&CK for Enterprise were derived from the all stages (recon, weaponize, delivery exploit, control, maintain, and execute) of a seven-stage Cyber Attack Lifecycle (first articulated by Lockheed Martin as the Cyber Kill Chain®). This provides a deeper level of granularity in describing what can occur during an intrusion."

https://attack.mitre.org/

Identifying the First Password Attempted in the Attack

Kill Chain Phase: Exploitation

What was the first brute force password used?

Hints

- Sourcetype that logs interactions between web browser and server is important
- Form_data has usernames and passwords in it
- The _time field shows when the event occurred
- Build on the search from the previous question

Adding Time

```
index=botsv1 sourcetype=stream:http form_data=*username*passwd*
| table _time form_data
```

All time

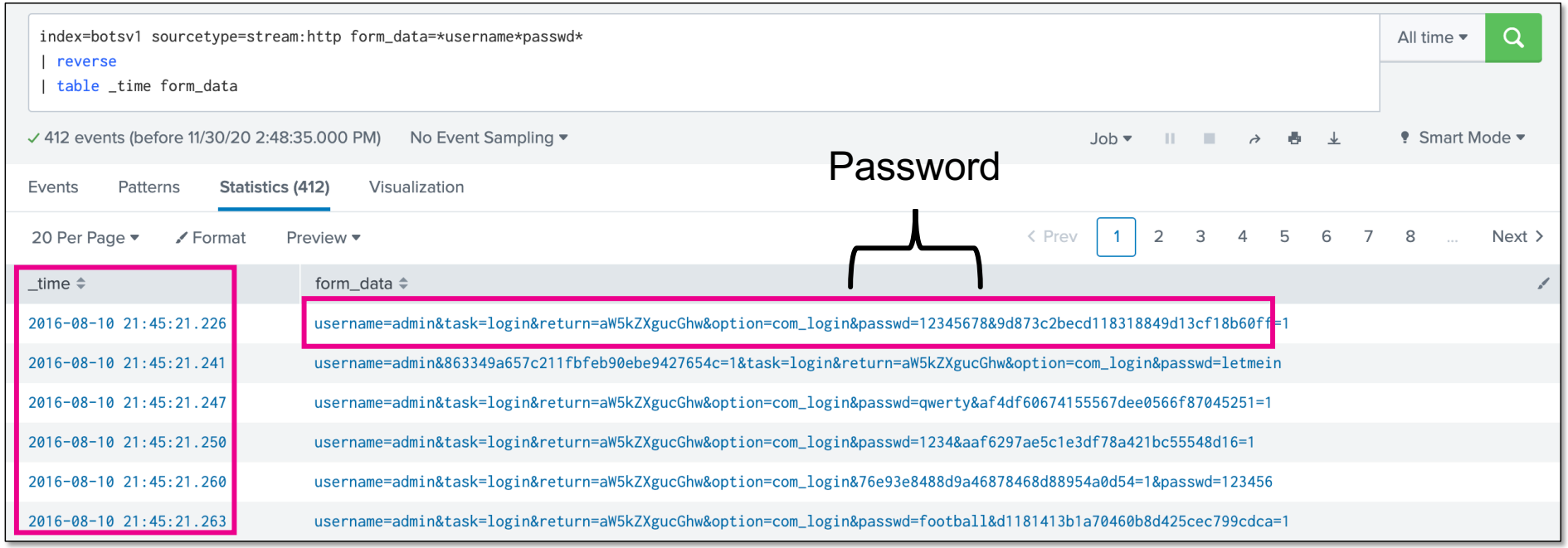
✓ 412 events (before 11/5/18 9:54:40.000 PM) No Event Sampling Job Smart Mode

Events Patterns **Statistics (412)** Visualization

20 Per Page Format Preview < Prev 1 2 3 4 5 6 7 8 ... Next >

_time	form_data
2016-08-10 21:48:05.858	username=admin&passwd=batman&option=com_login&task=login&return=aW5kZXgucGhw&e5ec827a3f67ce0efc546d81f7356acc=1
2016-08-10 21:46:51.394	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=rock&4a40c518220c1993f0e02dc4712c5794=1
2016-08-10 21:46:51.154	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=cool&a09349d0d6bdbf078ad72cf8e9348583=1
2016-08-10 21:46:51.156	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=sammy&0d3bb0020f70044ffba32f7d0fa7fa88=1
2016-08-10 21:46:50.873	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=august&9800c58b682f234e562dee5972a58b8d=1
2016-08-10 21:46:50.634	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=phantom&a083bf4d12c07976186d8a6efa6308cf=1
2016-08-10 21:46:50.627	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=williams&e3b1998d29669e83333a101735fd1c90=1
2016-08-10 21:46:50.621	username=admin&ba11501d963f628dfb862d3a07bbe674=1&task=login&return=aW5kZXgucGhw&option=com_login&passwd=private
2016-08-10 21:46:50.640	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=baby&26a9247d113c378cdf06f31fa2154f2c=1
2016-08-10 21:46:50.637	username=admin&task=login&return=aW5kZXgucGhw&option=com_login&passwd=dave&1b067a8762b4c8a9909ca68aae723e5a=1

Reversing the Order of Output (Oldest First)

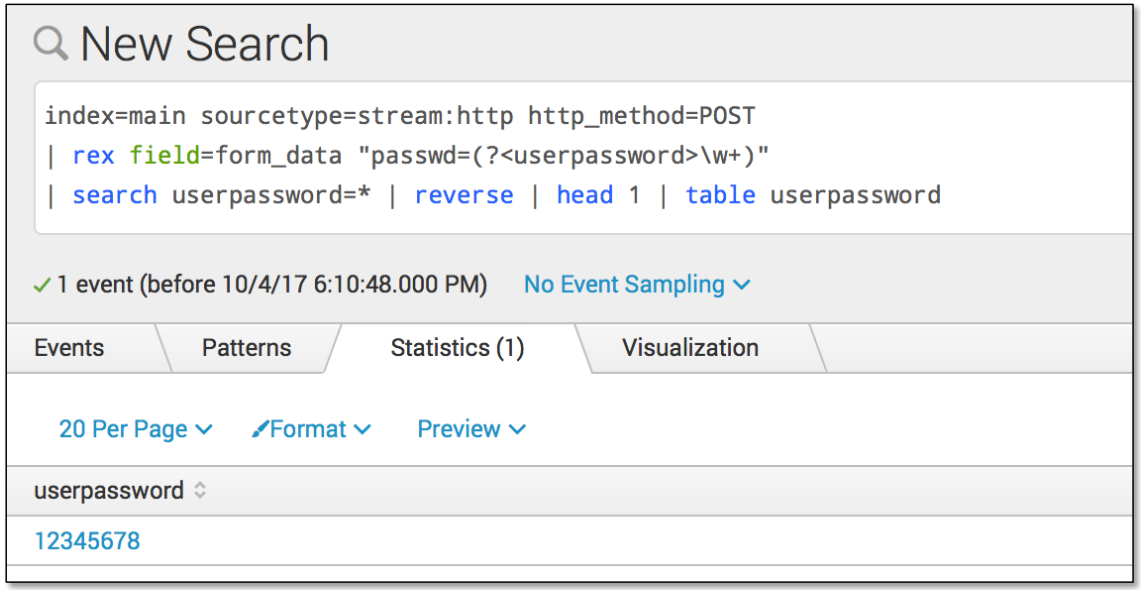


A More Elegant Way To View the Passwords

Kill Chain Phase: Exploitation

What was the first brute force password used?

- 12345678



Identifying the Password Used To Gain Access

Kill Chain Phase: Exploitation

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

Bonus Question: From what IP address was the password used?

Hints

- Sourcetype should be one that provides communication between web browser and server
- The stats command will be helpful to identify the correct password
- The command to extract the password from the form_data is

```
| rex field=form_data "passwd=(?<userpassword>\w+)"
```

What Else Do We Need?

```
index=botsv1 sourcetype=stream:http form_data=*username*passwd* dest_ip=192.168.250.70
| rex field=form_data "passwd=(?<userpassword>\w+)"
| stats count by userpassword
| sort - count
```

All time

✓ 412 events (before 11/5/18 11:57:28.000 PM) No Event Sampling Job Smart Mode

Events Patterns **Statistics (411)** Visualization

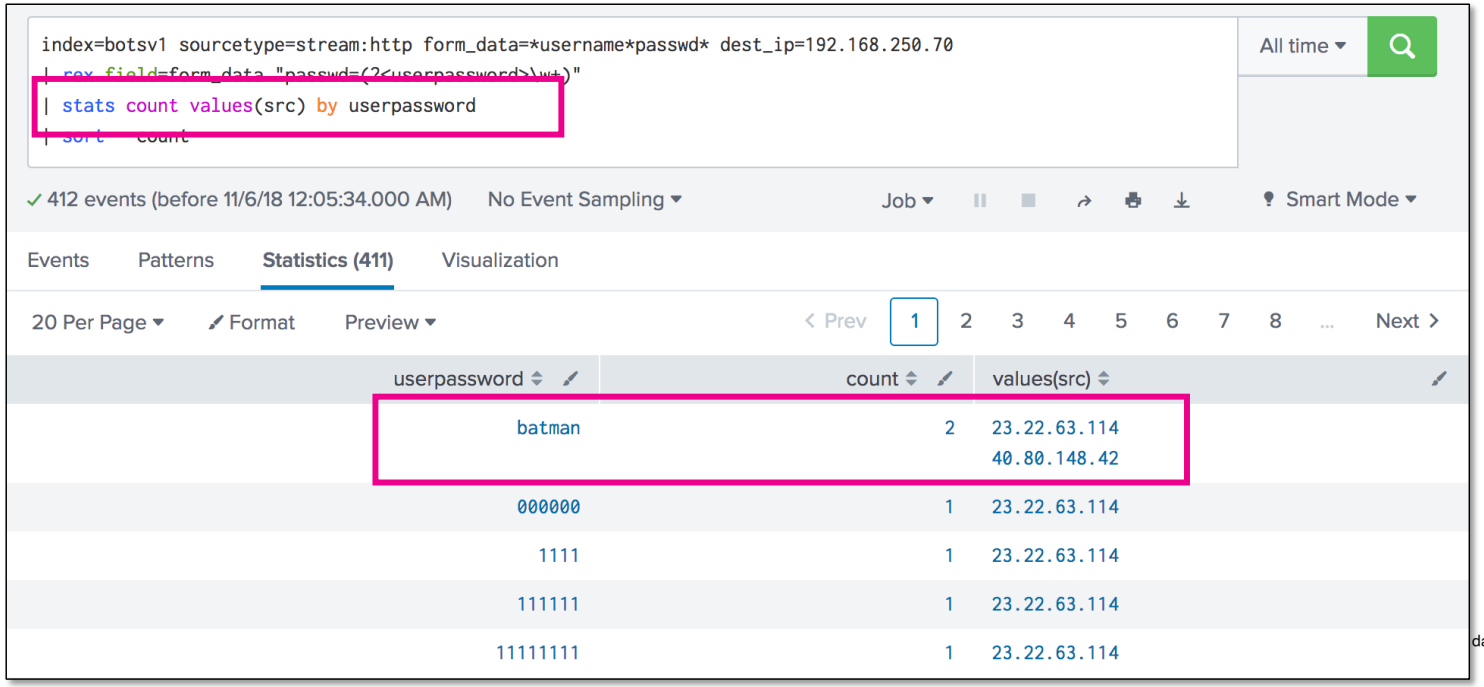
20 Per Page Format Preview < Prev 1 2 3 4 5 6 7 8 ... Next >

userpassword	count
batman	2
000000	1
1111	1
111111	1
11111111	1

- Focus our search on destination for greater precision
- Rex command will extract the passwords
- Add a stats command to count the number of times each password was used and sort by the count
- Second use of a password would likely indicate that it was the one the adversary wanted

stats Command Provides A Wider View

From an investigation perspective, knowing the password was used twice is nice, but knowing where it was used is far more interesting



data into doing

Collecting Additional Attributes Around the Login Events

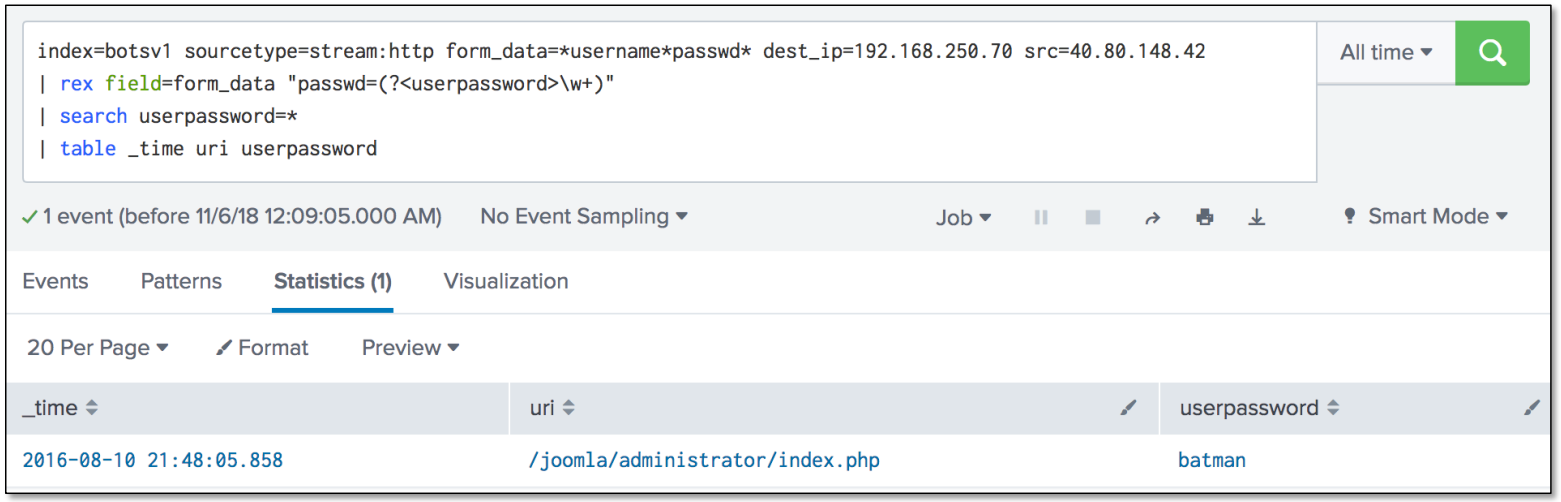
Kill Chain Phase: Exploitation

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

- batman

Bonus Question: From what IP address was the password used?

- 40.80.148.42

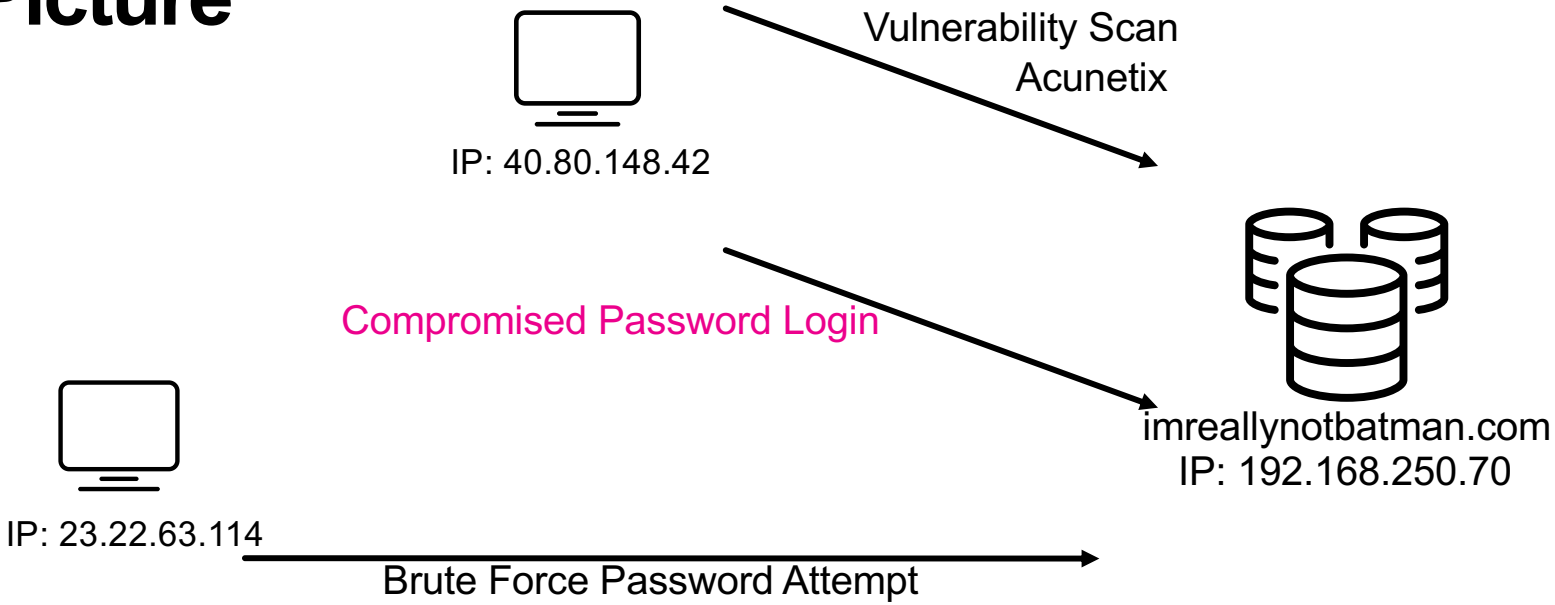


Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42

APT Picture



Determining The Elapsed Time Between Events

Kill Chain Phase: Exploitation

How many seconds elapsed between the time the brute force password scan identified the correct password and the compromised login?

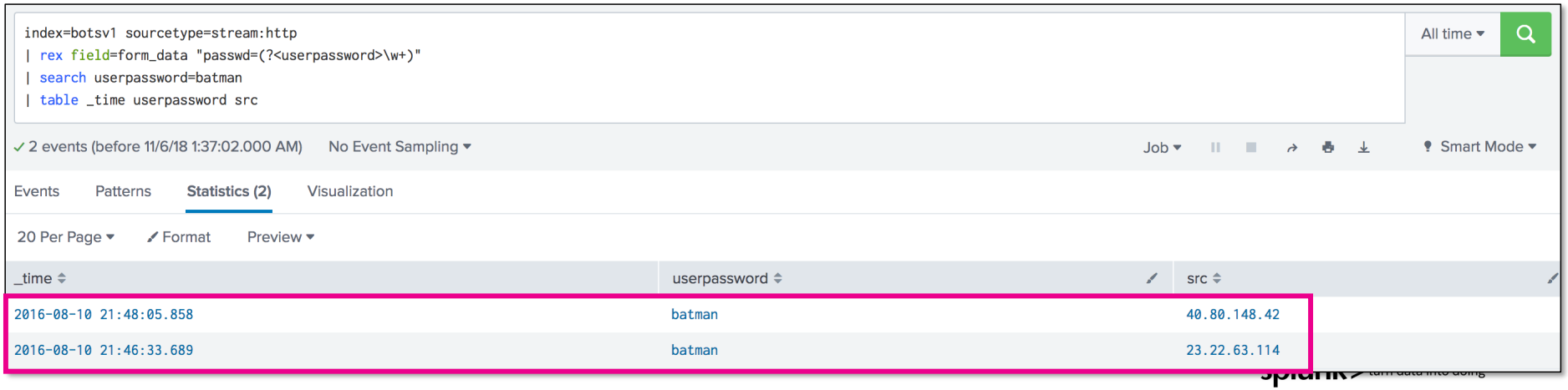
Hints

- Use the rex command that we used previously to extract the passwords
- Identify the password that was used multiple times
- Look at the transaction command

Tabling Our Logins with the Same Password

Build on our earlier searches for the userpassword extraction

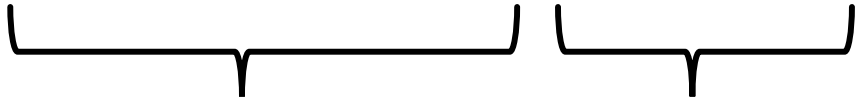
We know from earlier research the password of interest is batman, isolate search on those events



transaction command

Use transaction command to group these events together

```
index=main sourcetype=stream:http | rex field=form_data "passwd=(?<userpassword>\w+)"
| search userpassword=batman | transaction userpassword | table duration
```



Group events together based on a common value or values. In this case group events as a transaction based on the same userpassword

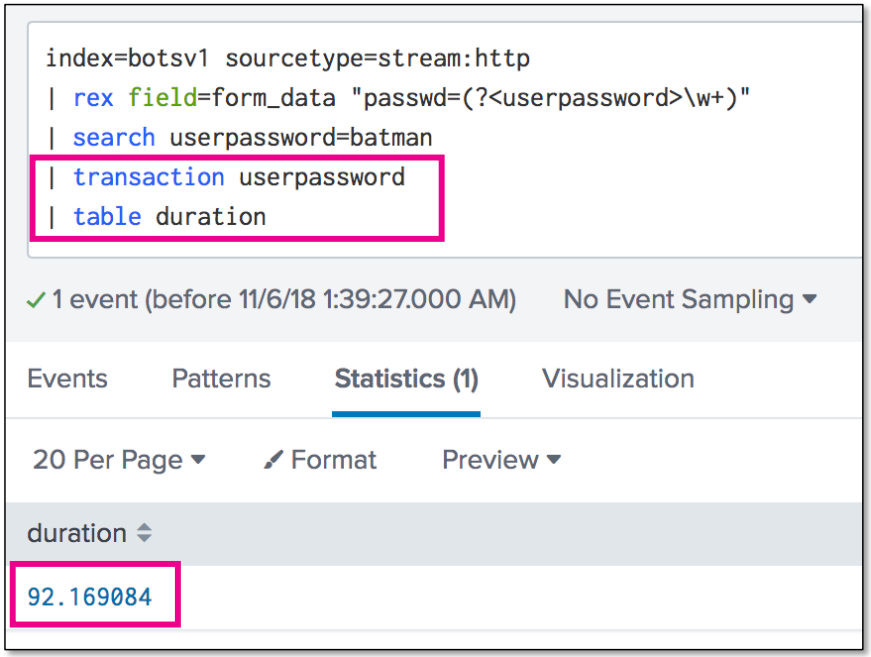
Duration is a value created with transaction that calculates the difference between the first and last event

Determining The Elapsed Time Between Events

Kill Chain Phase: Exploitation

How many seconds elapsed between the time the brute force password scan identified the correct password and the compromised login?

- 92.169084



Identifying the Executable Uploaded

Kill Chain Phase: Installation

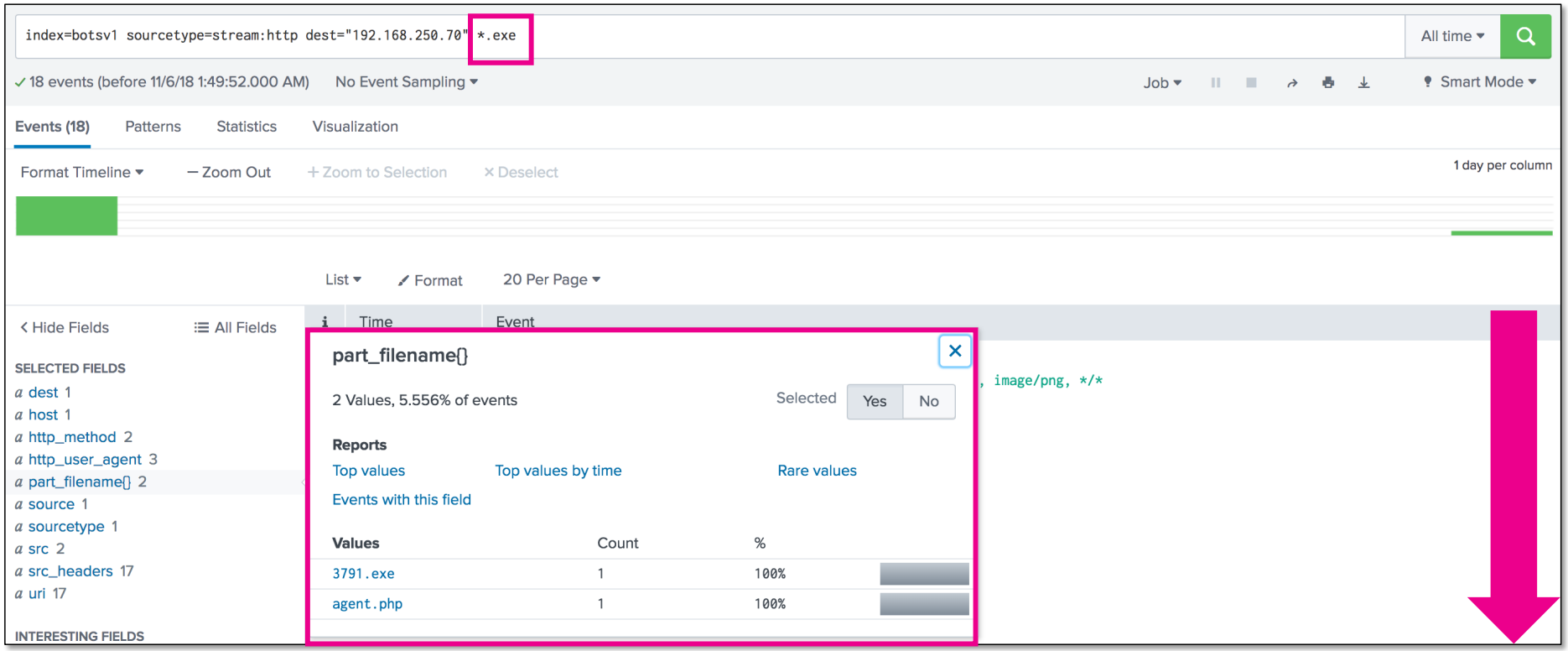
What is the name of the executable uploaded by P01s0n1vy?
(For example, "notepad.exe" or "favicon.ico")

Bonus Question: What is the source address of the executable?

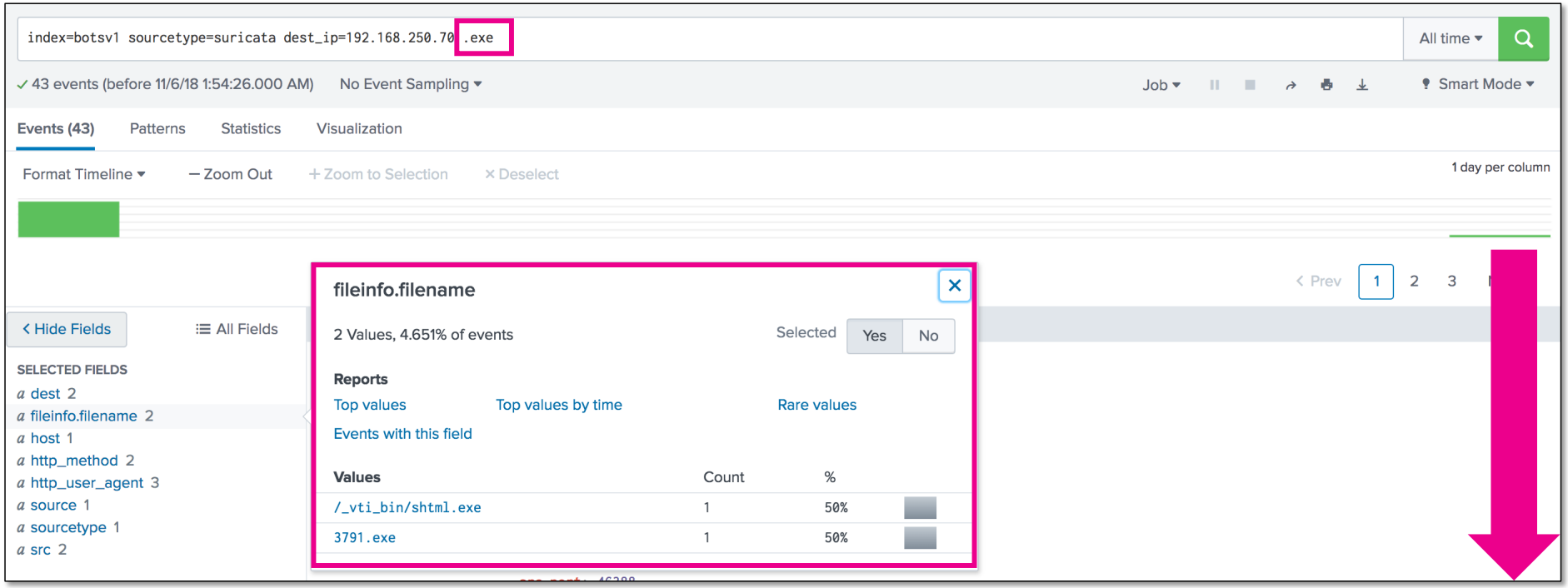
Hints

- If we are looking for executables uploaded, what sourcetypes would have visibility into files in motion?
- If a file is being uploaded what is likely to be the http_method?

Search for EXEs in stream:http



Search for EXEs in Suricata



Hostnames v IPs

Important Note About Fields

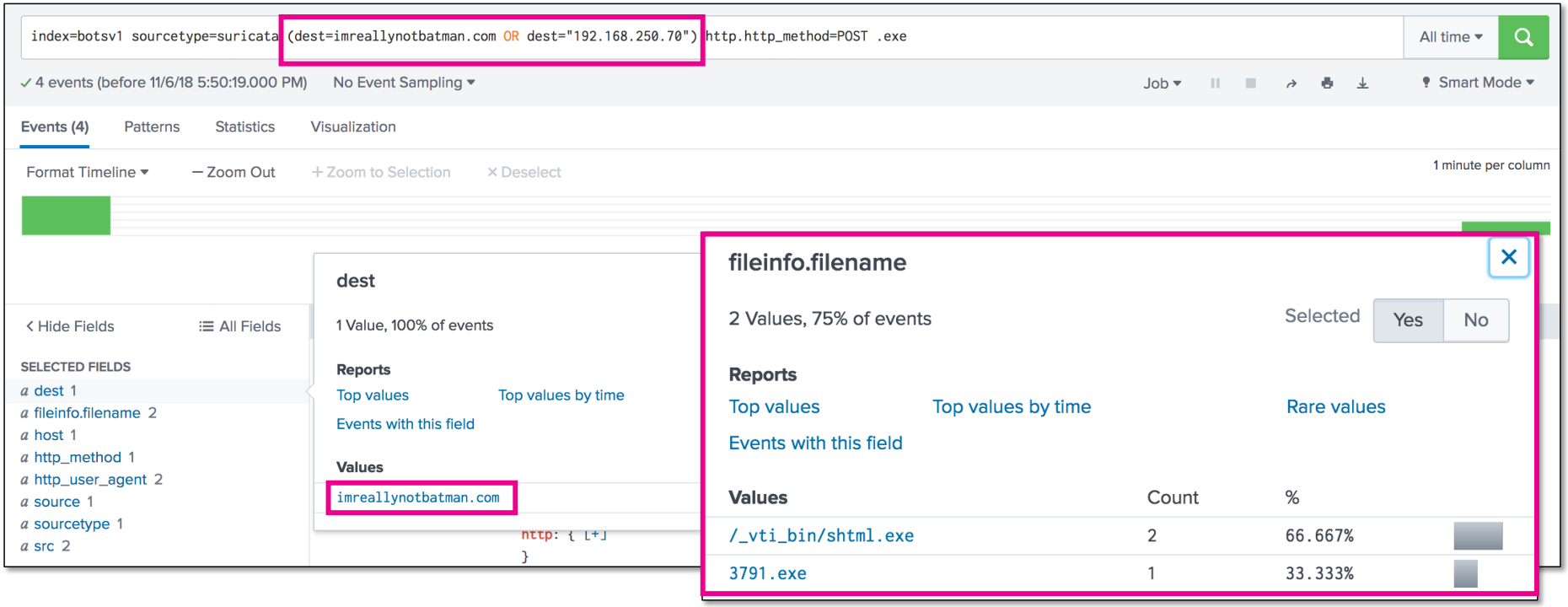
Fields may not have the same values based on the logging structure

dest v dest_ip could have different values

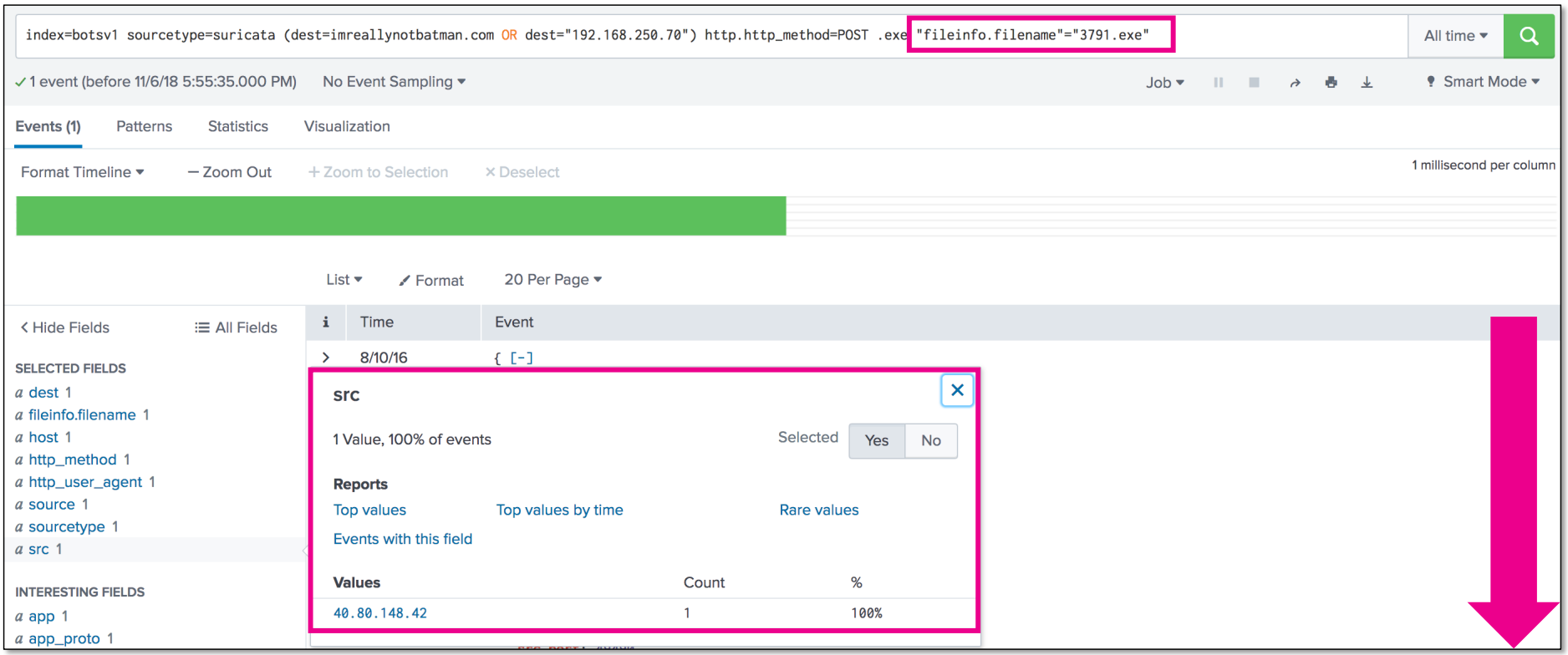
- Try using OR and parenthesis to bracket the search (dest=1.1.1.1 OR dest_ip=1.1.1.1)

```
index=botsv1 sourcetype=suricata (dest="192.168.250.70" OR dest_ip="192.168.250.70") .exe
```

When Destination and Destination IP are Different



Capturing the Source of the Executable



Identifying the Executable Uploaded

Kill Chain Phase: Installation

What is the name of the executable uploaded by P01s0n1vy? Please include file extension. (For example, "notepad.exe" or "favicon.ico")?

- 3791.exe

Bonus Question: What is the source of the executable?

- 40.80.148.82

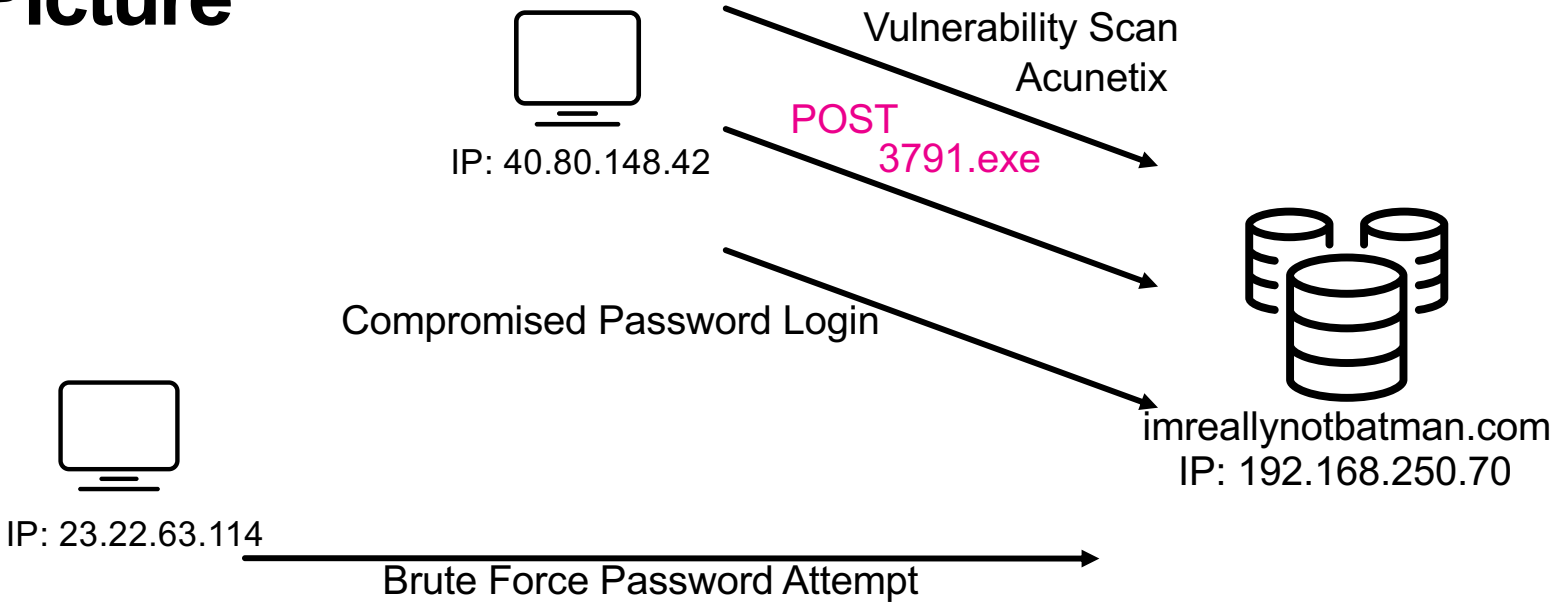
Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Executable Uploaded: 3791.exe

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

APT Picture



Determining the Hash of the Uploaded File

Kill Chain Phase: Installation

What is the MD5 hash of the executable uploaded?

Hints

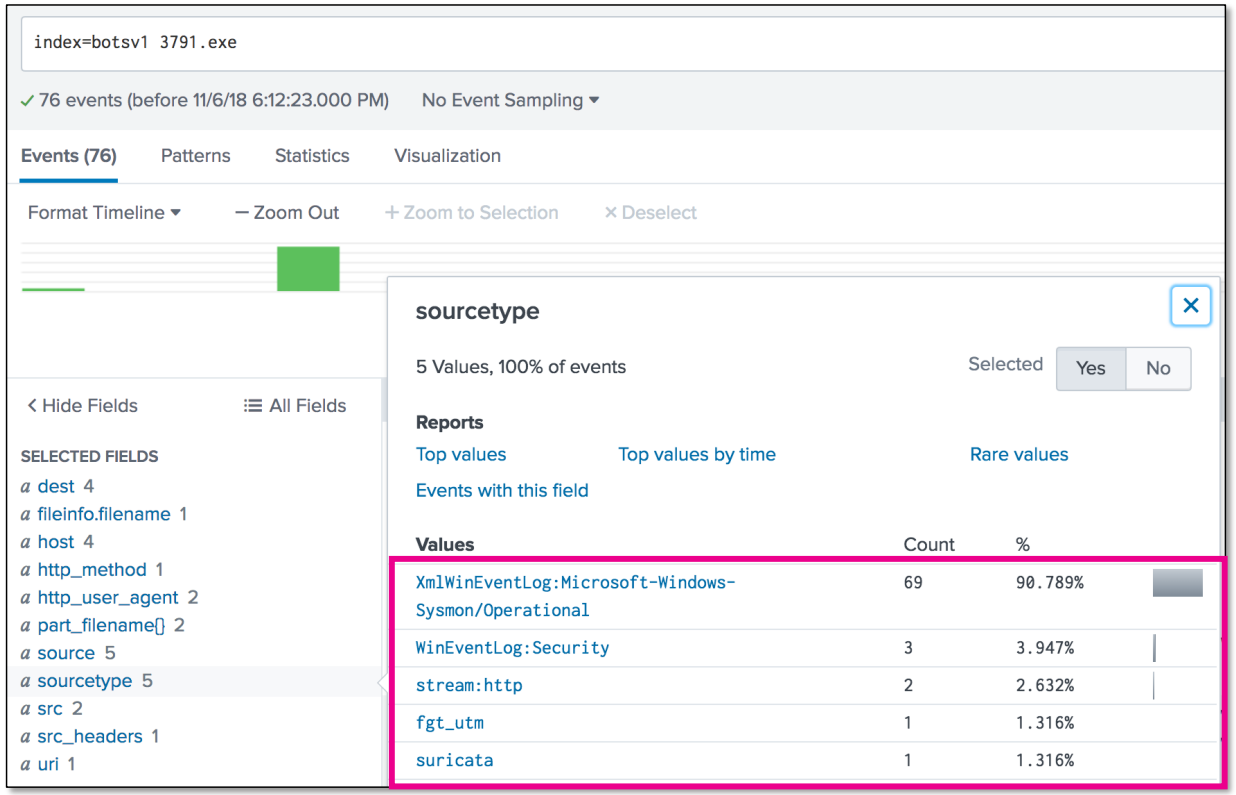
- Which sourcetypes have the correct filename?
- File hashes can be found in the log stream if you know which sourcetype to look in...
- Process Execution events will capture the MD5 at process launch

What sourcetype Should I Start With?

Overview of Sysmon Capabilities

- Records the hash of process image files using SHA1 (the default), MD5, SHA256 or IMPHASH.

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



sysmon Primer

Windows System Service and Device Driver

Resident across reboots

Logs numerous Windows activities including

- Process Creation
- Network Connection
- Driver Loaded
- File Creation
- Registry Event
- And more...

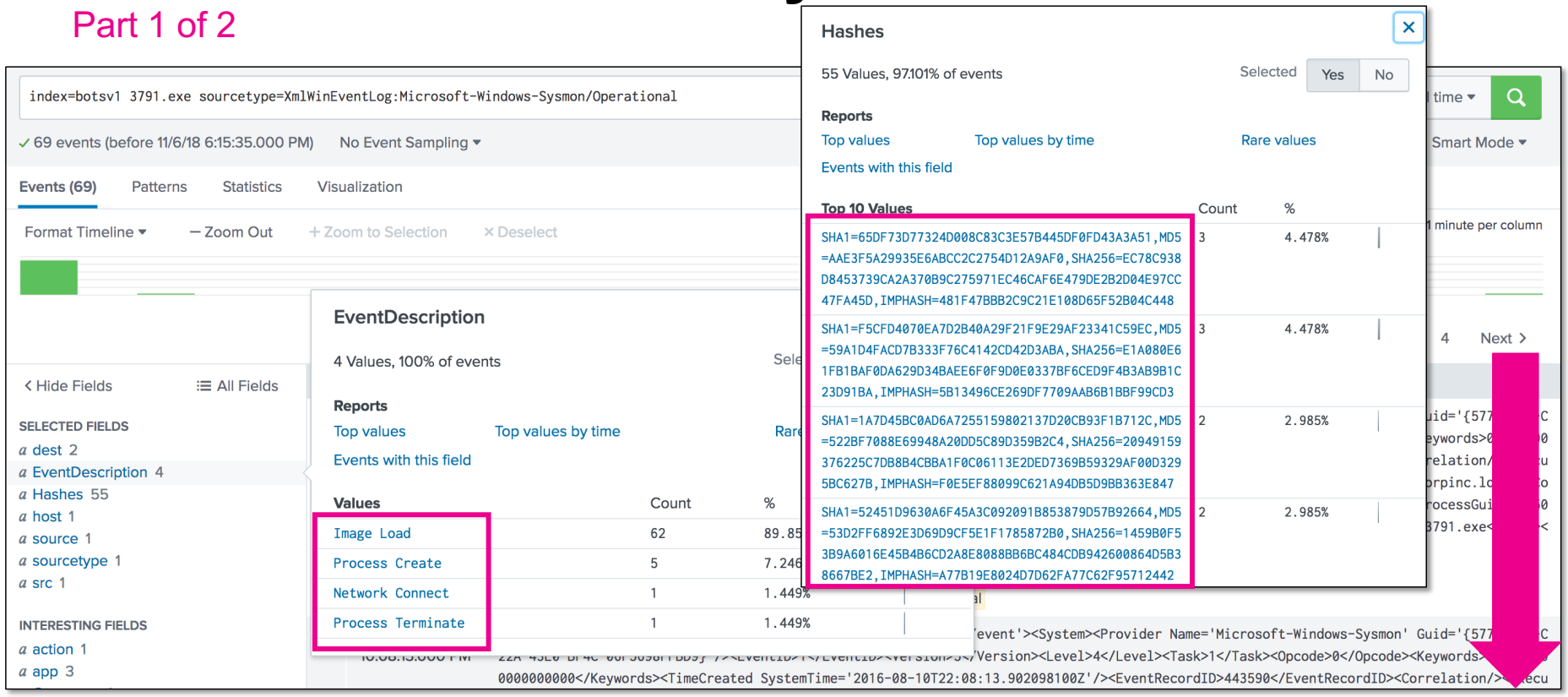
```
<Sysmon  schemaversion="3.2">
  <!-- Capture all the hashes -->
  <HashAlgorithms>*</HashAlgorithms>
  <EventFiltering>
    <!-- Log all drivers except if the signature -->
    <!-- contains Microsoft or Windows -->
    <DriverLoad onmatch="exclude">
      <Signature condition="contains">microsoft</Signature>
      <Signature condition="contains">windows</Signature>
    </DriverLoad>
    <!-- Do not log process termination -->
    <ProcessTerminate onmatch="include"/>
    <!-- Log network connection if the destination port equals 443 -->
    <!-- or 80, and the process isn't InternetExplorer -->
    <NetworkConnect onmatch="include">
      <DestinationPort>443</DestinationPort>
      <DestinationPort>80</DestinationPort>
    </NetworkConnect>
    <NetworkConnect onmatch="exclude">
      <Image condition="end with">iexplore.exe</Image>
    </NetworkConnect>
  </EventFiltering>
</Sysmon>
```

Configuration files in XML are used to determine which events are logged

- https://github.com/SwiftOnSecurity/sysmon-config

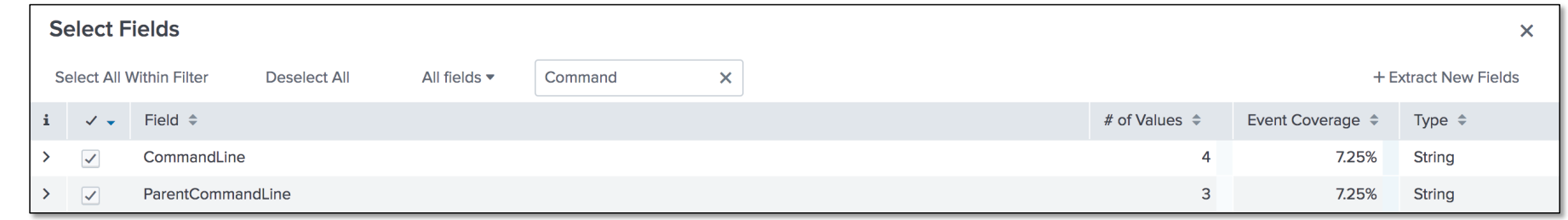
What Can I Find With Sysmon?

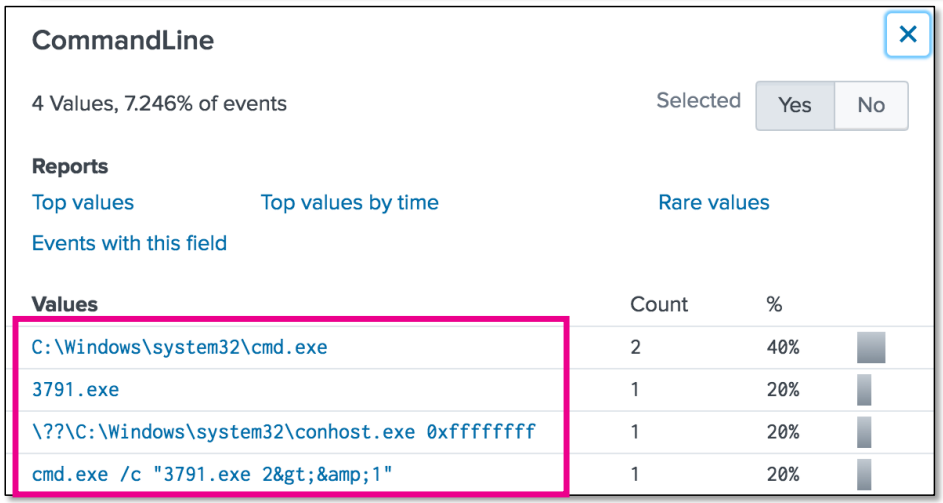
Part 1 of 2

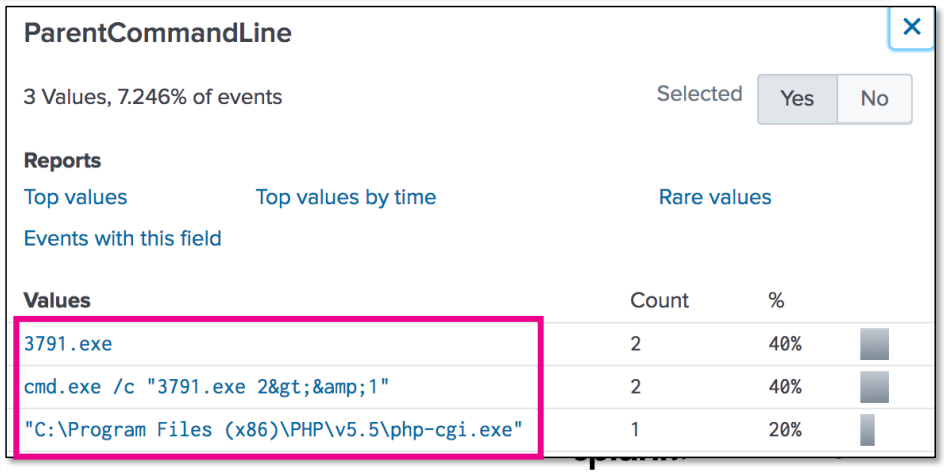


What Can I Find With Sysmon?

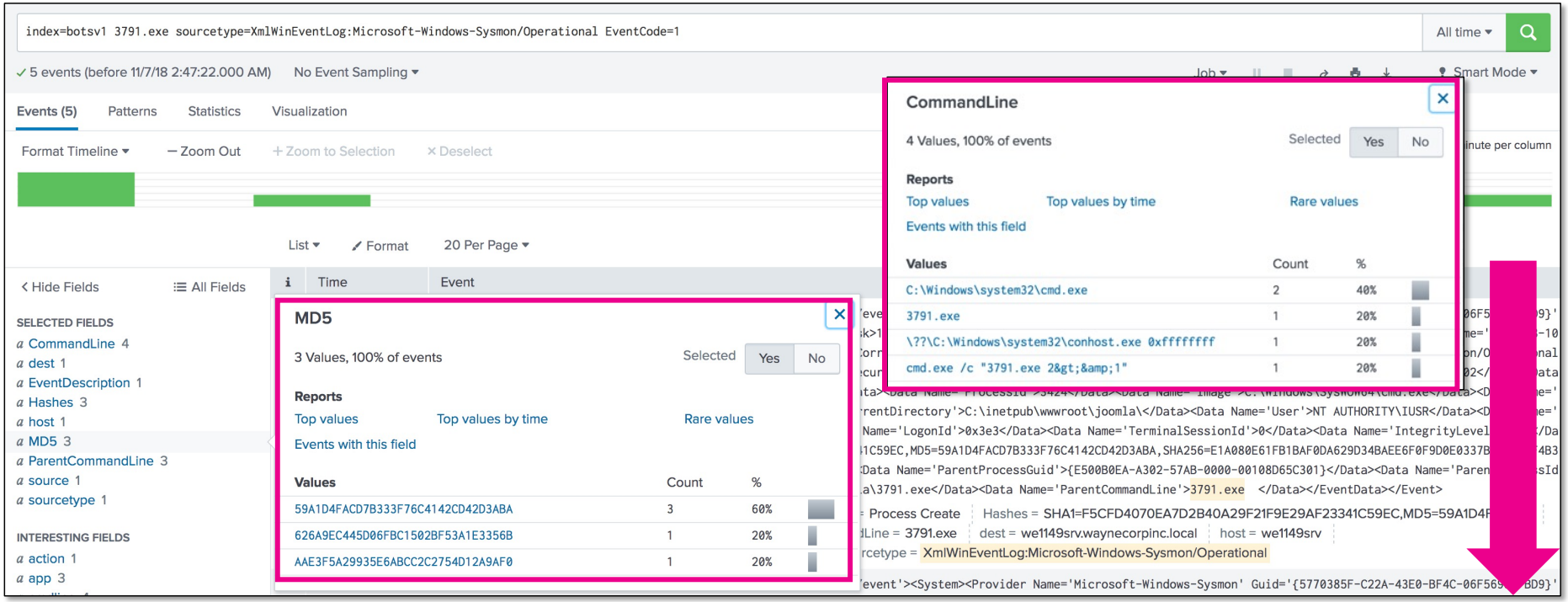
Part 2 of 2







Drilling Down on MD5 & CommandLine



splunk> turn data into doing

Exploring the Sysmon Event

```
index=botsv1 3791.exe sourcetype=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational EventCode=1 CommandLine="3791.exe"
```

8/10/16
9:56:18.000 PM

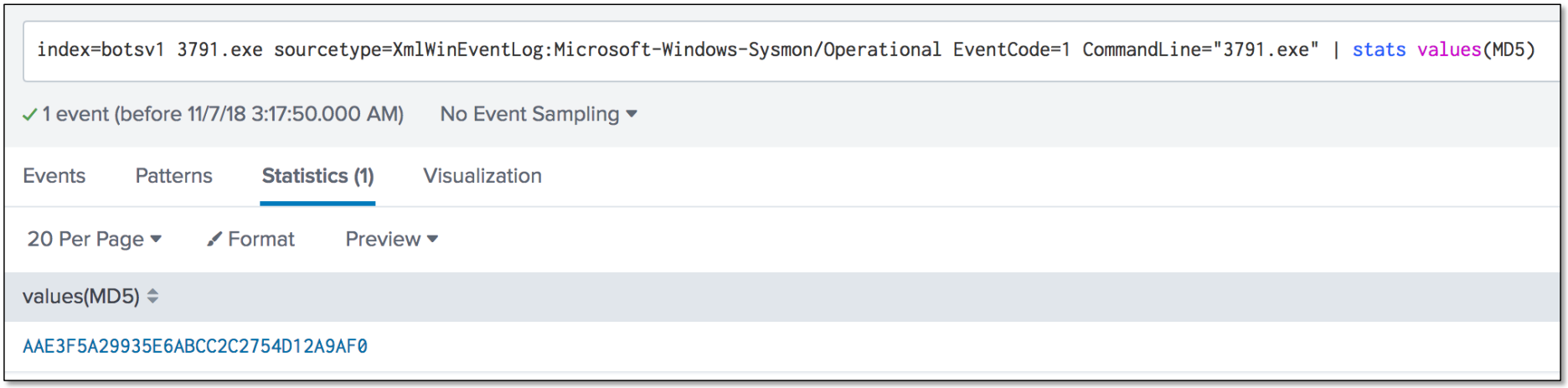
<Event xmlns='http://schemas.microsoft.com/win/2004/08/events/event'><System><Provider Name='Microsoft-Windows-Sysmon' Guid='{5770385F-C22A-43E0-BF4C-06F5698FFBD9}' /><EventID>1</EventID><Version>5</Version><Level>4</Level><Task>1</Task><Opcode>0</Opcode><Keywords>0x8000000000000000</Keywords><TimeCreated SystemTime='2016-08-10T21:56:18.142461700Z'/><EventRecordID>428908</EventRecordID><Correlation/><Execution ProcessID='1296' ThreadID='1416'/><Channel>Microsoft-Windows-Sysmon/Operational</Channel><Computer>we1149srv.waynecorpinc.local</Computer><Security UserID='S-1-5-18'/></System><EventData><Data Name='UtcTime'>2016-08-10 21:56:18.142</Data><Data Name='ProcessGuid'>{E500B0EA-A302-57AB-0000-00108D65C301}</Data><Data Name='ProcessId'>3880</Data><Data Name='Image'>C:\inetpub\wwwroot\joomla\3791.exe</Data><Data Name='CommandLine'>3791.exe </Data><Data Name='CurrentDirectory'>C:\inetpub\wwwroot\joomla\</Data><Data Name='User'>NT AUTHORITY\IUSR</Data><Data Name='LogonGuid'>{E500B0EA-219E-57AA-0000-0020E3030000}</Data><Data Name='LogonId'>0x3e3</Data><Data Name='TerminalSessionId'>0</Data><Data Name='IntegrityLevel'>High</Data><Data Name='Hashes'>SHA1=65DF73D77324D008C83C3E57B445DF0FD43A3A51,MD5=AAE3F5A29935E6ABCC2C2754D12A9AF0,SHA256=EC78C938D8453739CA2A370B9C275971EC46CAF6E479DE2B2D04E97CC47FA45D,IMPHASH=481F47BBB2C9C21E108D65F52B04C448</Data><Data Name='ParentProcessGuid'>{E500B0EA-A302-57AB-0000-00102E63C301}</Data><Data Name='ParentProcessId'>2896</Data><Data Name='ParentImage'>C:\Windows\SysWOW64\cmd.exe</Data><Data Name='ParentCommandLine'>cmd.exe /c "3791.exe 2>&1"</Data></EventData></Event>

Event Actions

Type	Field	Value	Actions
Selected	CommandLine	3791.exe	
	EventDescription	Process Create	
	Hashes	SHA1=65DF73D77324D008C83C3E57B445DF0FD43A3A51,MD5=AAE3F5A29935E6ABCC2C2754D12A9AF0,SHA256=EC78C938D8453739CA2A370B9C275971EC46CAF6E479DE2B2D04E97CC47FA45D,IMPHASH=481F47BBB2C9C21E108D65F52B04C448	
	MD5	AAE3F5A29935E6ABCC2C2754D12A9AF0	
	ParentCommandLine	cmd.exe /c "3791.exe 2>&1"	
	dest	we1149srv.waynecorpinc.local	
	host	we1149srv	
	source	WinEventLog:Microsoft-Windows-Sysmon/Operational	
	sourcetype	XmlWinEventLog:Microsoft-Windows-Sysmon/Operational	
Event	Computer	we1149srv.waynecorpinc.local	
	CurrentDirectory	C:\inetpub\wwwroot\joomla\	
	EventChannel	Microsoft-Windows-Sysmon/Operational	
	EventCode	1	

to doing

Putting it Together



Determining the Hash of the Uploaded File

Kill Chain Phase: Installation

What is the MD5 hash of the executable uploaded?

- AAE3F5A29935E6ABCC2C2754D12A9AF0

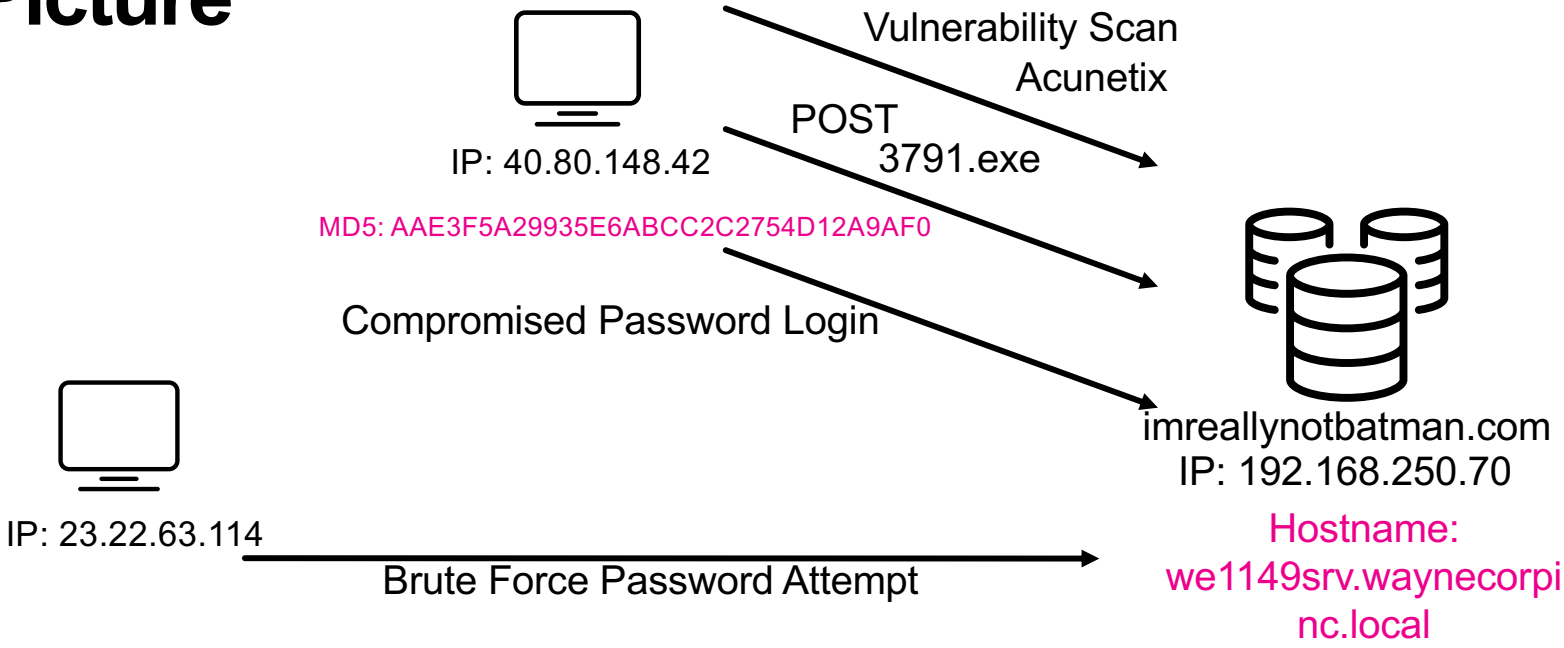
Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Executable Uploaded: 3791.exe
Hash of the Uploaded File:
AAE3F5A29935E6ABCC2C2754D12A9AF0

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

APT Picture



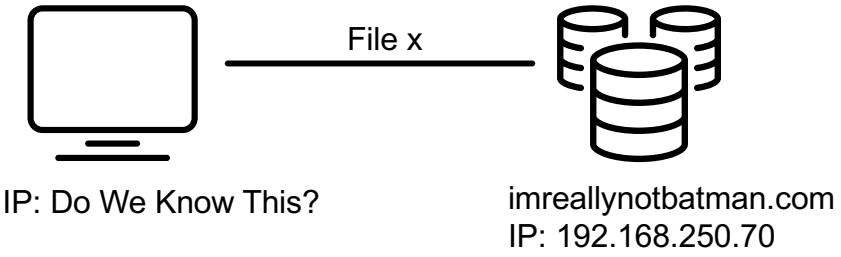
Identifying the File that Defaced Our Web Server

Kill Chain Phase: Actions on Objective

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

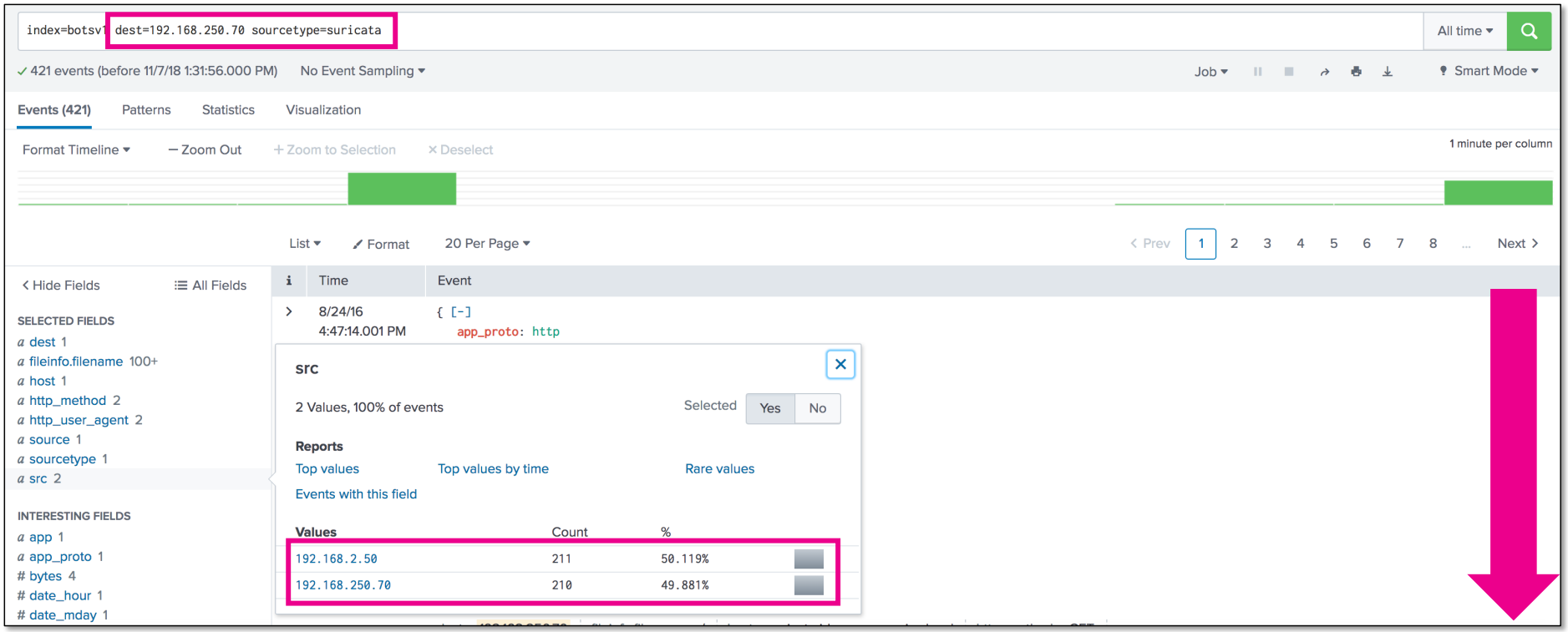
Hints

- Data sources that show communication between the web browser and server will provide good visibility
- Determining the IP address of the other system could be helpful
- The flow of information (directionality) is the important key.
  - POST v GET
  - Previous information gathered in the investigation will be helpful here



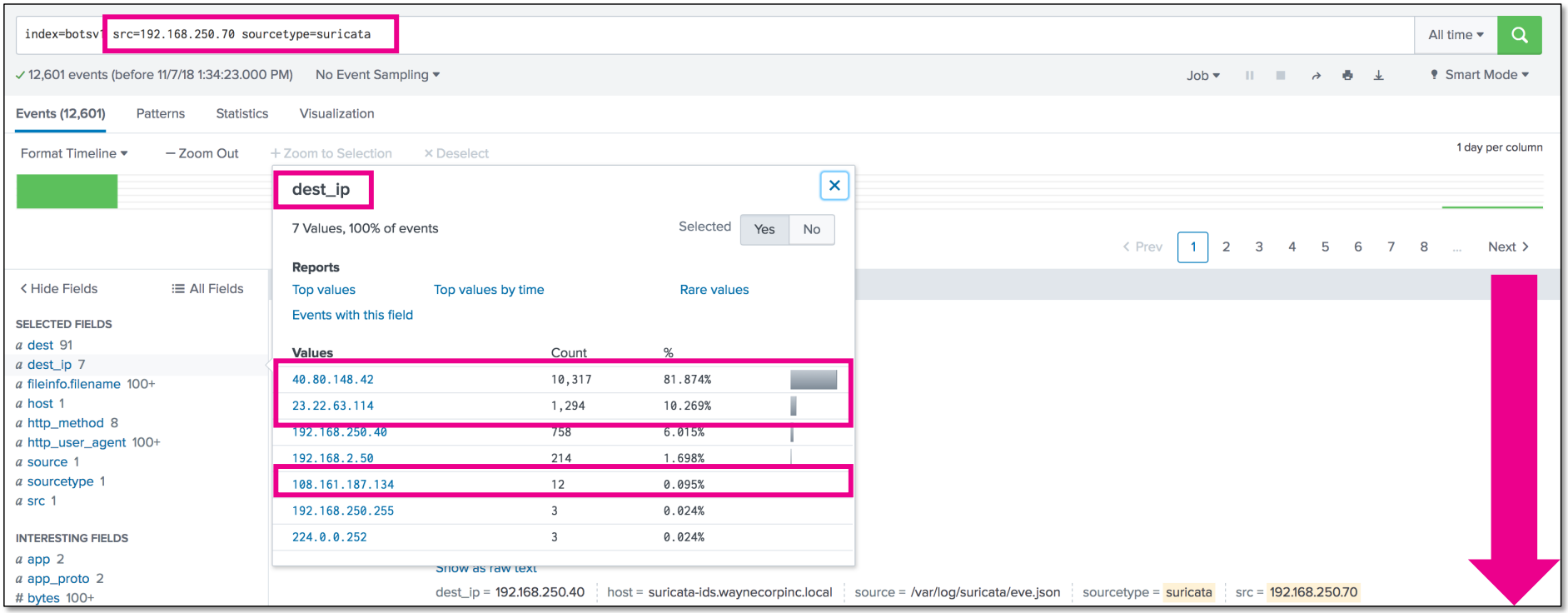
Looking at Directional Flow of Data - Part 1

Nothing Coming From Outside In

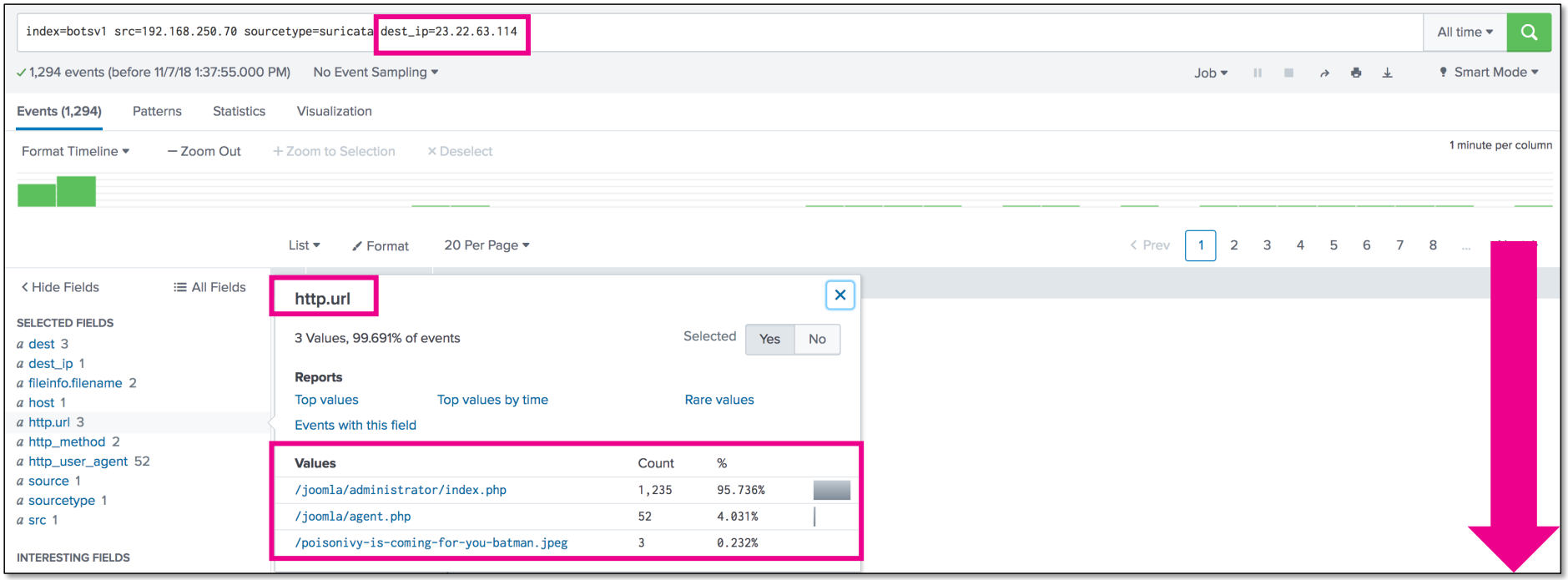


Looking at Directional Flow of Data - Part 2

Looking from Inside Out

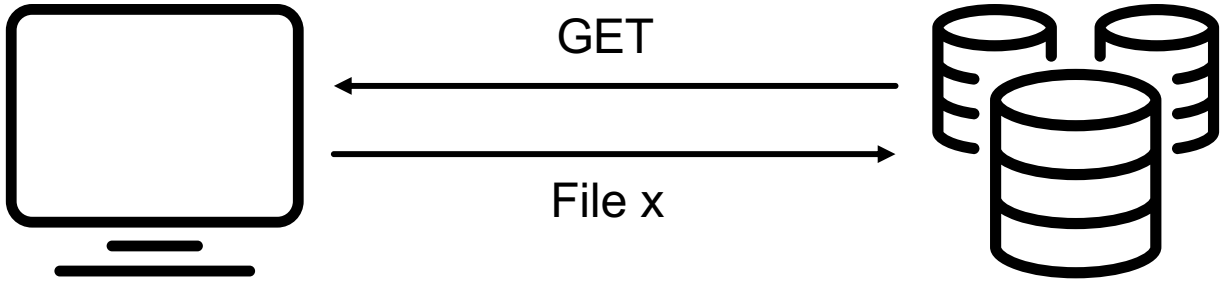


Pivot Into Destination IP Addresses to View URLs



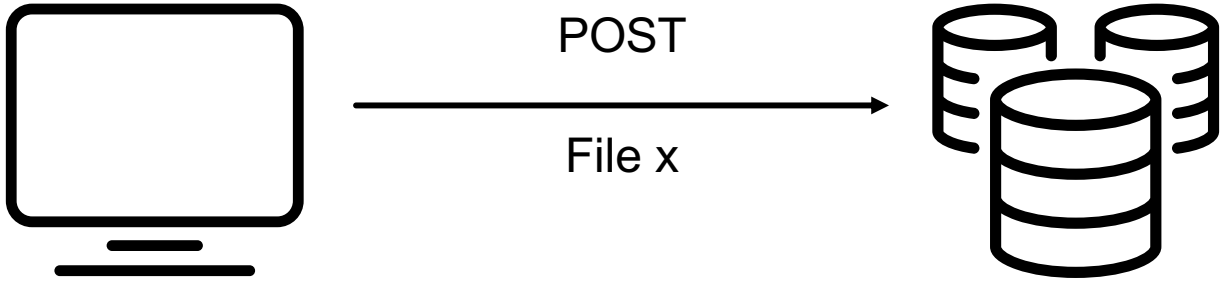
HTTP Get v Post Traffic Flow

HTTP GET will generally retrieve data



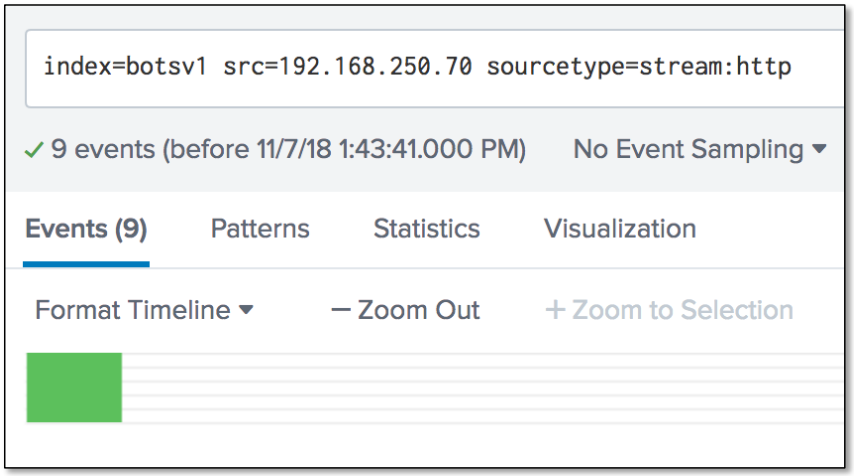
imreallynotbatman.com

HTTP POST submits data to be processed



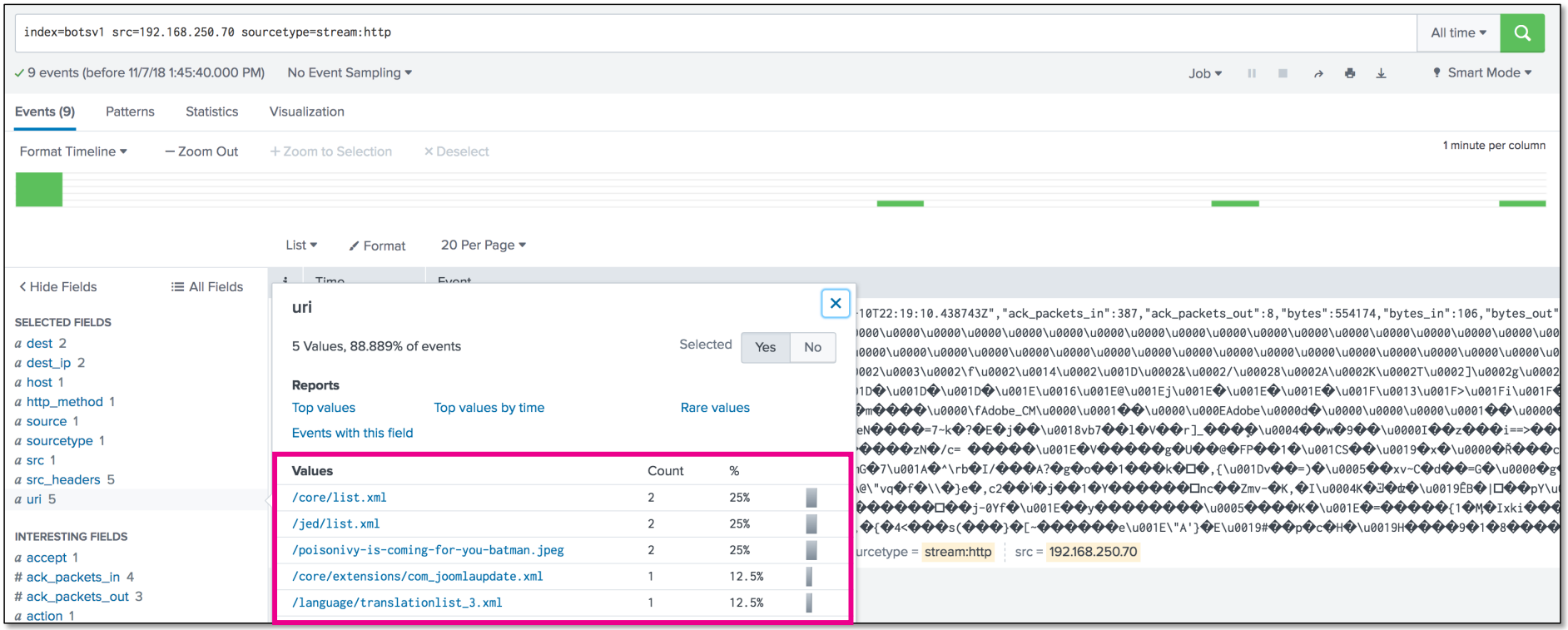
Do Web Servers Start A Conversation?

How often should web servers start a conversation?

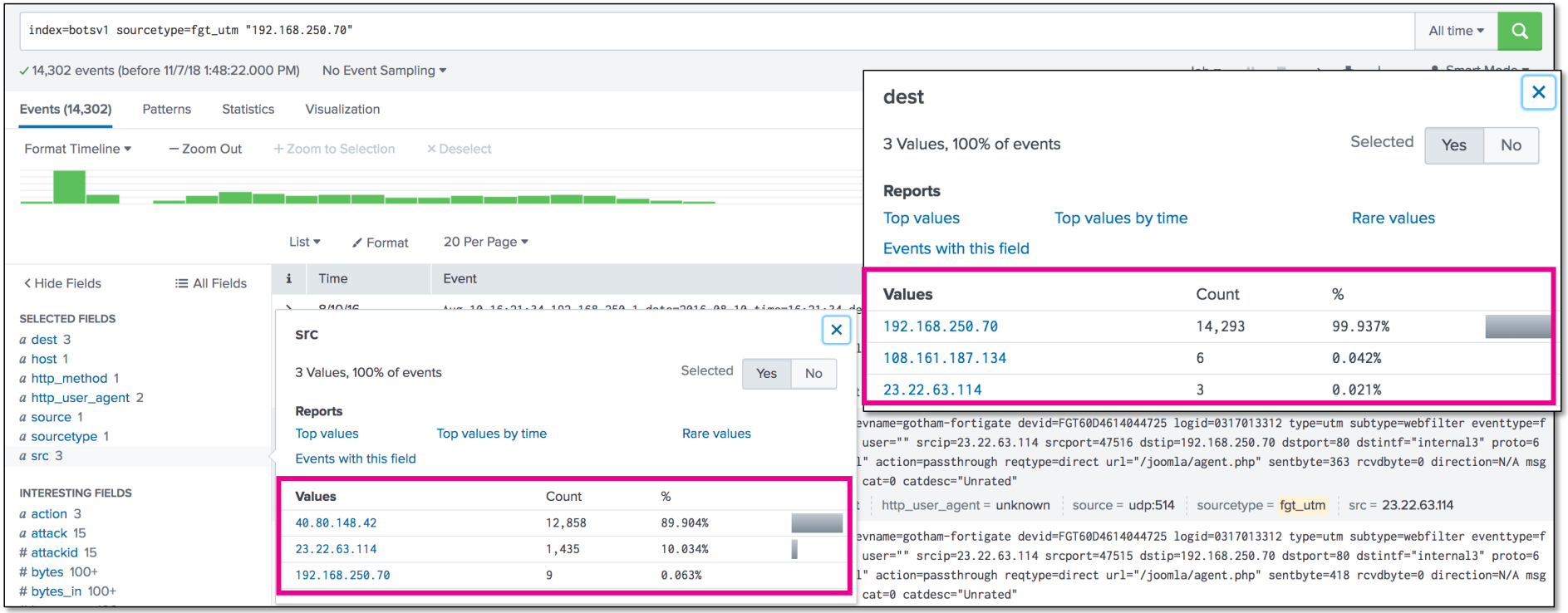


If they are starting the conversation is that interesting and should be looked at?

Any Similarities between Suricata and stream:http?



What About Firewall Data?



Which Search Do I Start With?

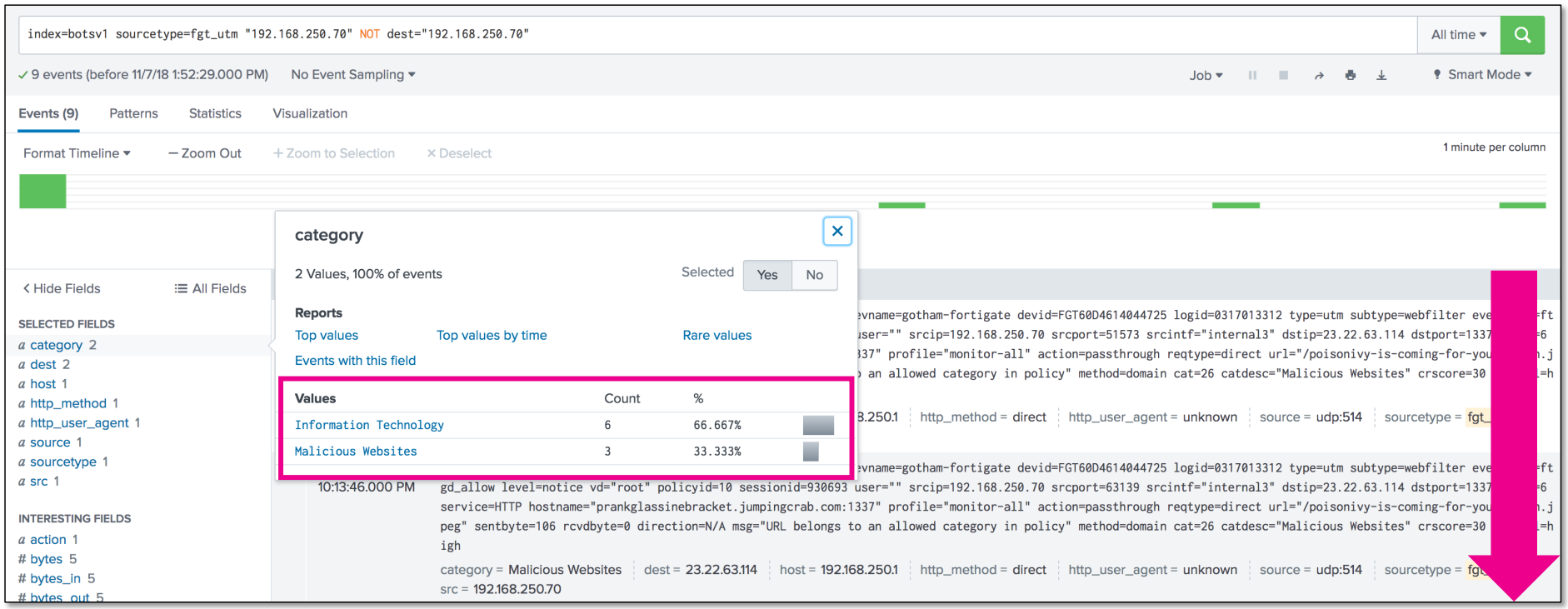
```
index=botsv1 sourcetype=fgt_utm "192.168.250.70" NOT dest="192.168.250.70" | stats count
```

✓ 9 events (before 11/7/18 1:50:52.000 PM) No Event Sampling ▾

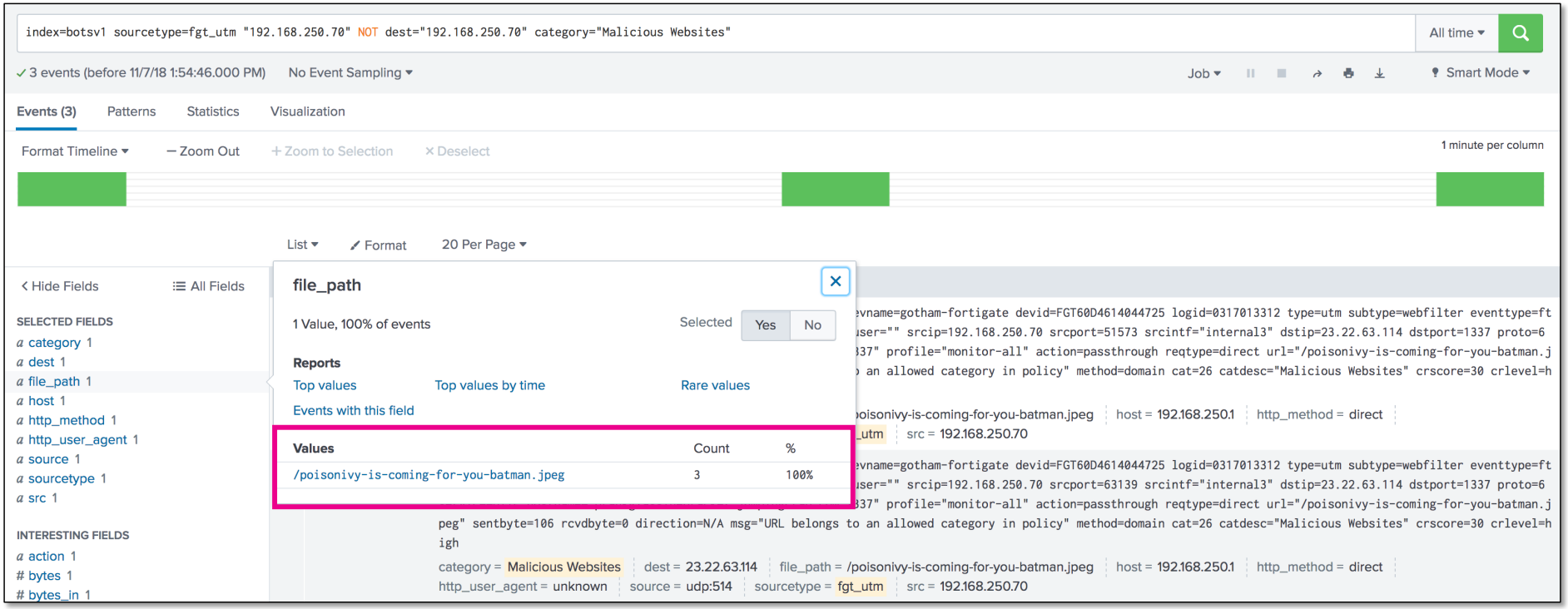
```
index=botsv1 sourcetype=fgt_utm "192.168.250.70" NOT src="192.168.250.70" |stats count
```

✓ 14,293 events (before 11/7/18 1:50:55.000 PM) No Event Sampling ▾

Use Web Site Categorization to Filter



Firewall Gives Us Confirmation



Identifying the File that Defaced Our Web Server

Kill Chain Phase: Actions on Objective

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

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

Kill Chain

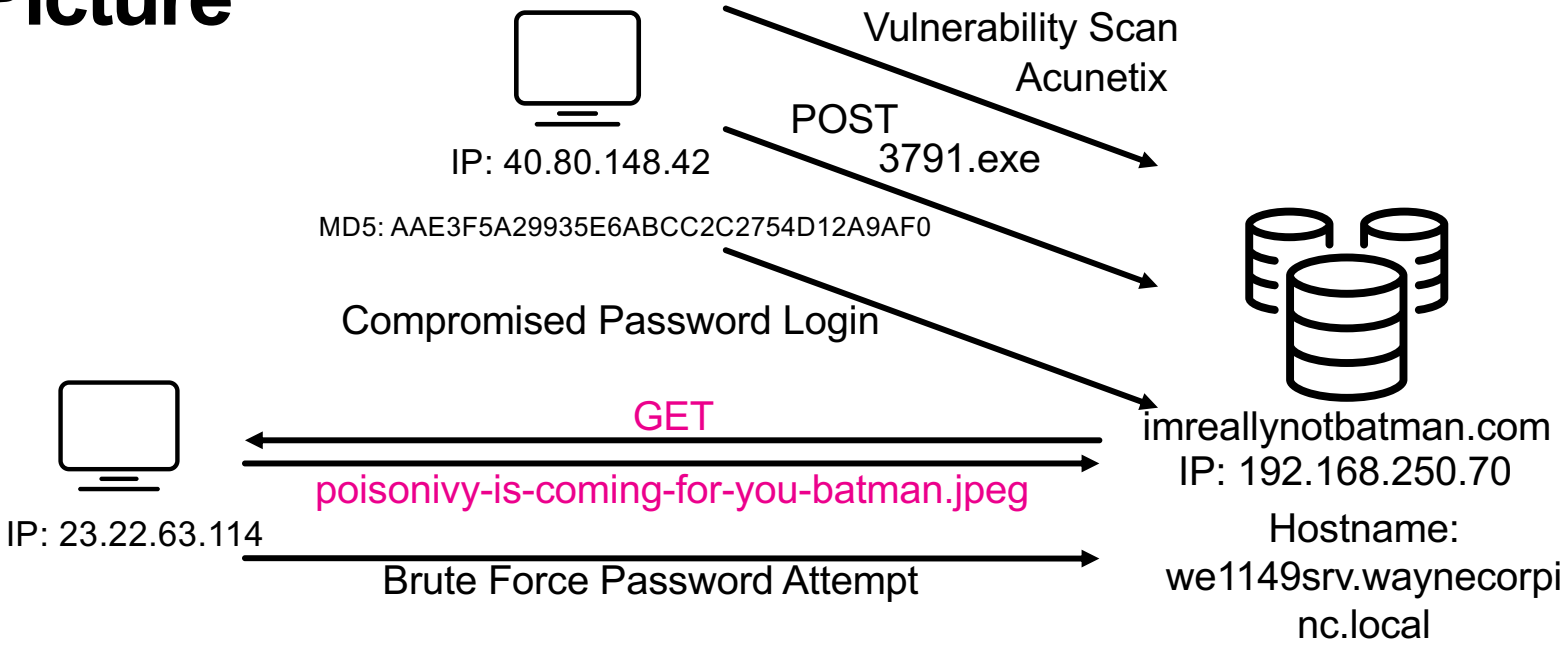
IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Executable Uploaded: 3791.exe
Hash of the Uploaded File: AAE3F5A29935E6ABCC2C2754D12A9AF0

File that Defaced Our Web Server: poisonivy-is-coming-for-you-batman.jpeg

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

APT Picture



Identifying the FQDN of the System that Defaced...

Kill Chain Phase: Command & Control

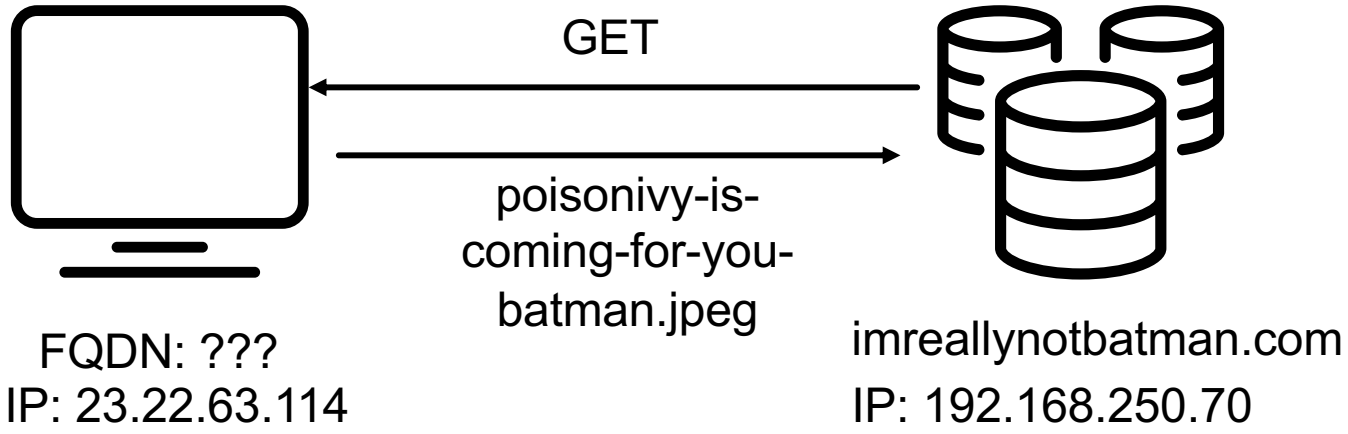
This attack used dynamic DNS to resolve to the malicious IP. What fully qualified domain name (FQDN) is associated with this attack?

Hint

- Information just recently uncovered will help answer this

HTTP Get Traffic Flow

HTTP GET will generally retrieves data



Using Fortigate Firewall Events We Just Found

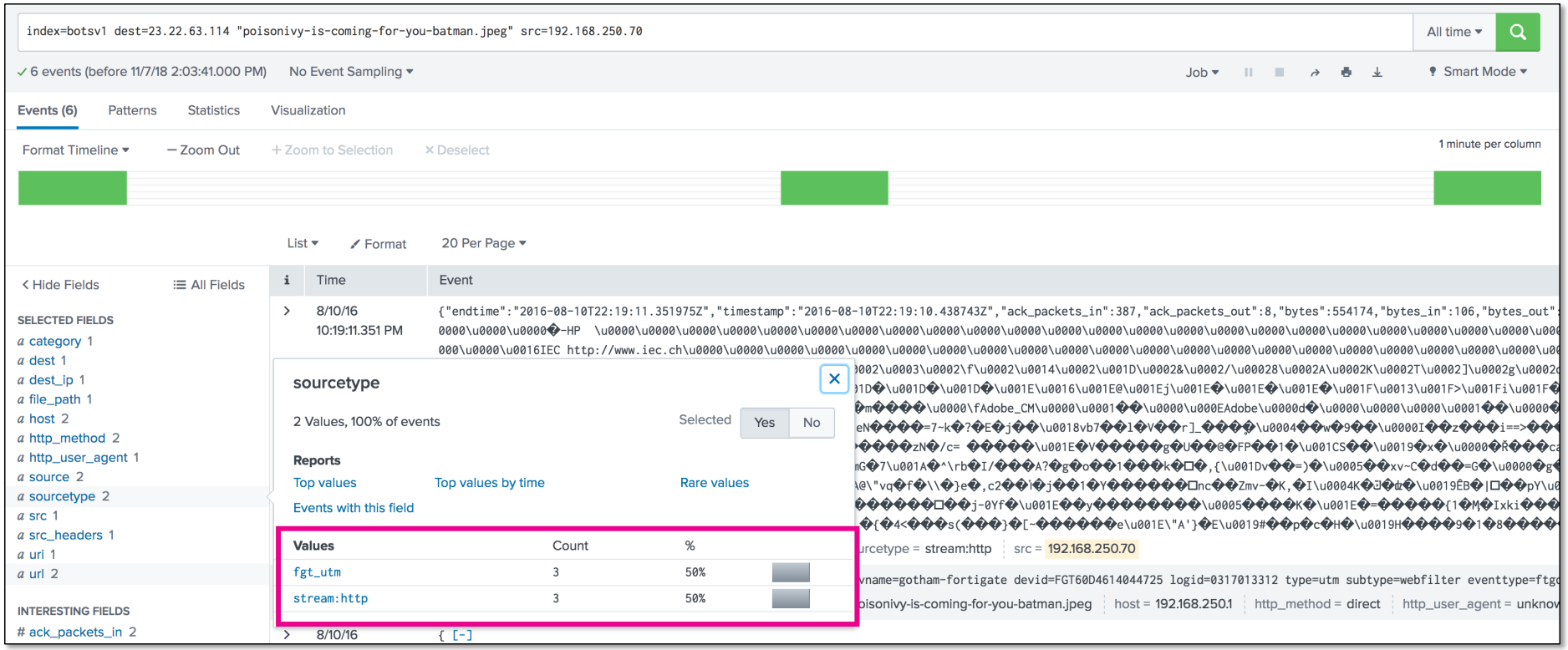
8/10/16
10:19:10.000 PM

```
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 subtype=webfilter eventtype=ftgd_allow level=notice vd="root" policyid=10 sessionid=932526 user="" srcip=192.168.250.70 srcport=51573 srcintf="internal3" dstip=23.22.63.114 dstport=1337 proto=6 service=HTTP hostname="prankglassinebracket.jumpingcrab.com:1337" profile="monitor-all" action=passthrough reqtype=direct url="/poisonivy-is-coming-for-you-batman.jpeg" sentbyte=106 rcvdbyte=0 direction=N/A msg="URL belongs to an allowed category in policy" method=domain cat=26 catdesc="Malicious Websites" crscore=30 crlevel=high
```

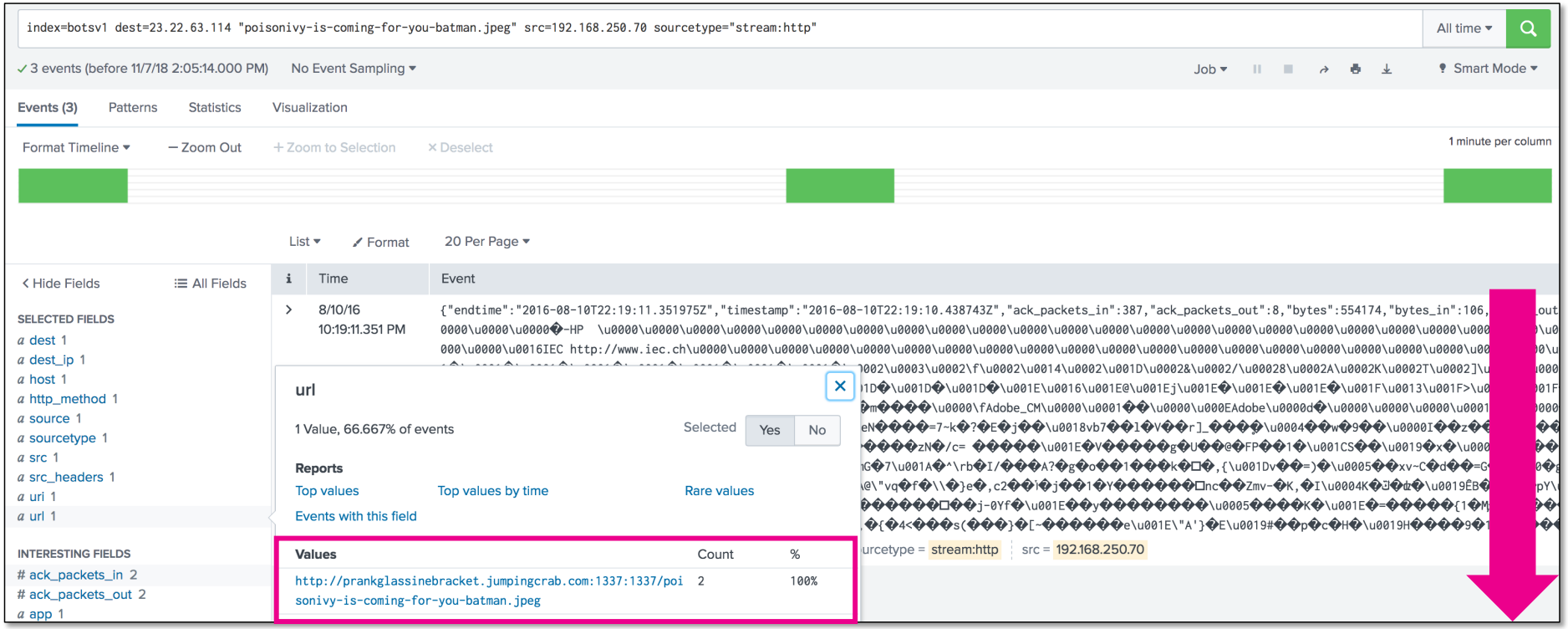
Event Actions

Type	Field	Value	Actions
Selected	category	Malicious Websites	
	dest	23.22.63.114	
	file_path	/poisonivy-is-coming-for-you-batman.jpeg	
	host	192.168.250.1	
	http_method	direct	
	http_user_agent	unknown	
	source	udp:514	
	sourcetype	fgt_utm	
	src	192.168.250.70	
	url	prankglassinebracket.jumpingcrab.com:1337/poisonivy-is-coming-for-you-batman.jpeg	
Event	action	allowed	

What Data Sets Saw This File?



Using stream:http



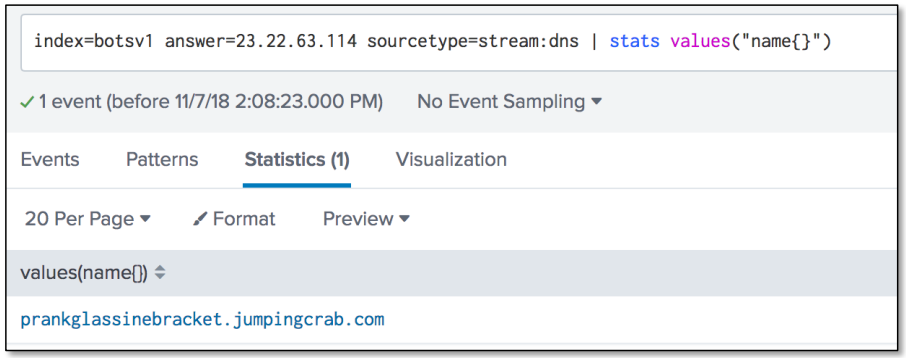
splunk> turn data into doing

What if I Don't Have the File Name?

If we know there was DNS resolution to the malicious IP, how can we craft a search?

DNS will be associated with the stream:dns sourcetype

We identified a couple of IP addresses of interest even if we don't have the filename, we probably know that 23.22.63.114 is of concern



Identifying the FQDN of the System that Defaced...

Kill Chain Phase: Command & Control

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

- prankglassinebracket.jumpingcrab.com

More information on this topic

- http://blogs.splunk.com/2015/08/04/detecting-dynamic-dns-domains-in-splunk/

Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Executable Uploaded: 3791.exe
Hash of the Uploaded File: AAE3F5A29935E6ABCC2C2754D12A9AF0

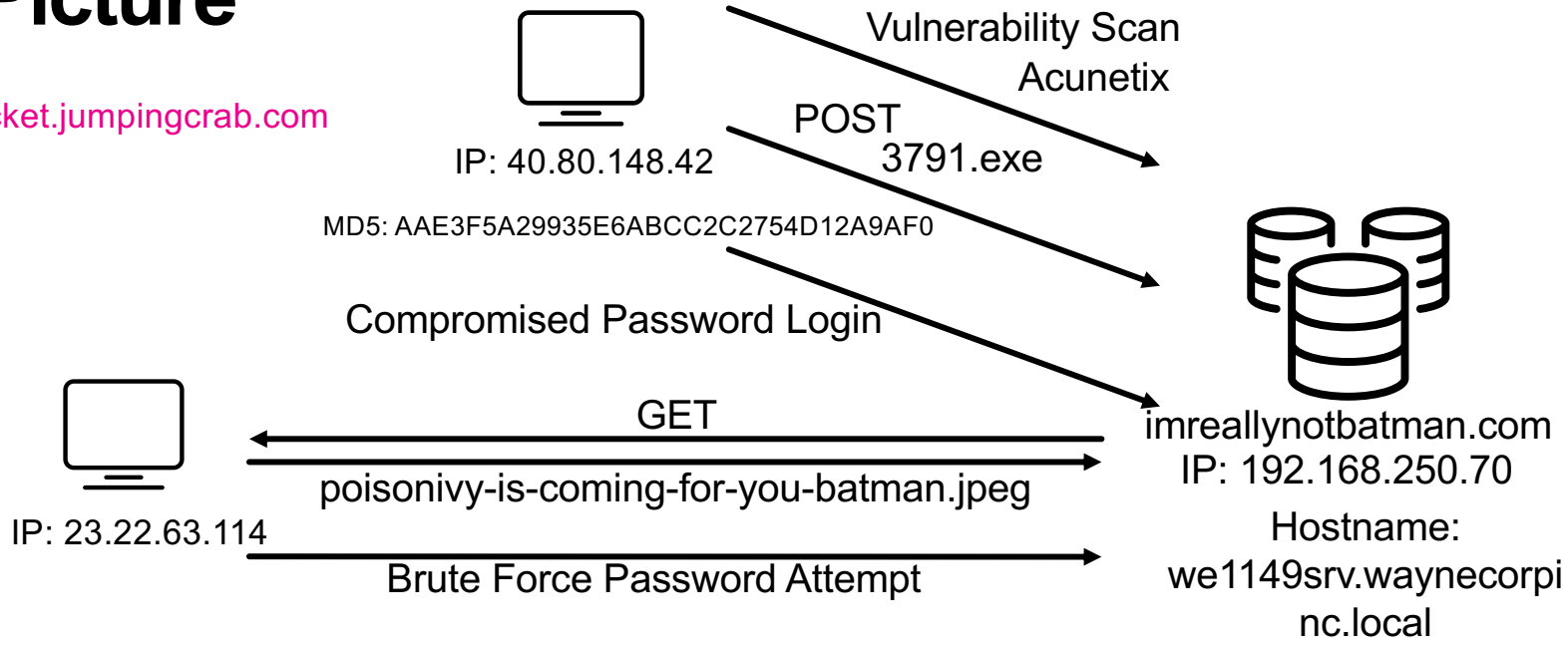
File that Defaced Our Web Server: poisonivy-is-coming-for-you-batman.jpeg

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

Fully Qualified Domain Name of the System that Defaced the Web Server: prankglassinebracket.jumpingcrab.com

APT Picture

DNS:
prankglassinebracket.jumpingcrab.com



Using OSINT to Identify Attacker Infrastructure

Kill Chain Phase: Weaponization

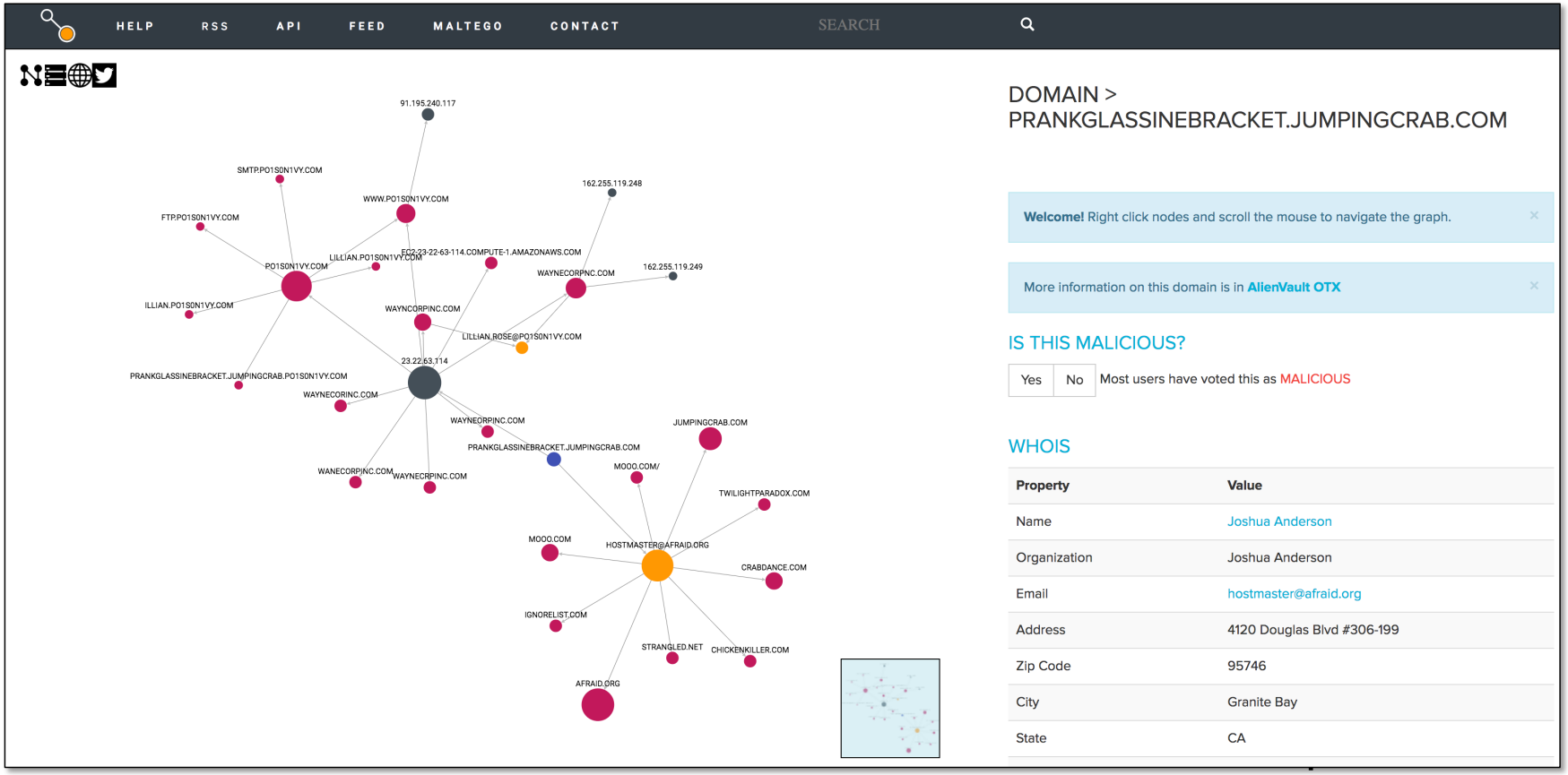
What IP address has P01s0n1vy tied to domains that are pre-staged to attack Wayne Enterprises?

Bonus Question: What additional domains were tied to that IP address?

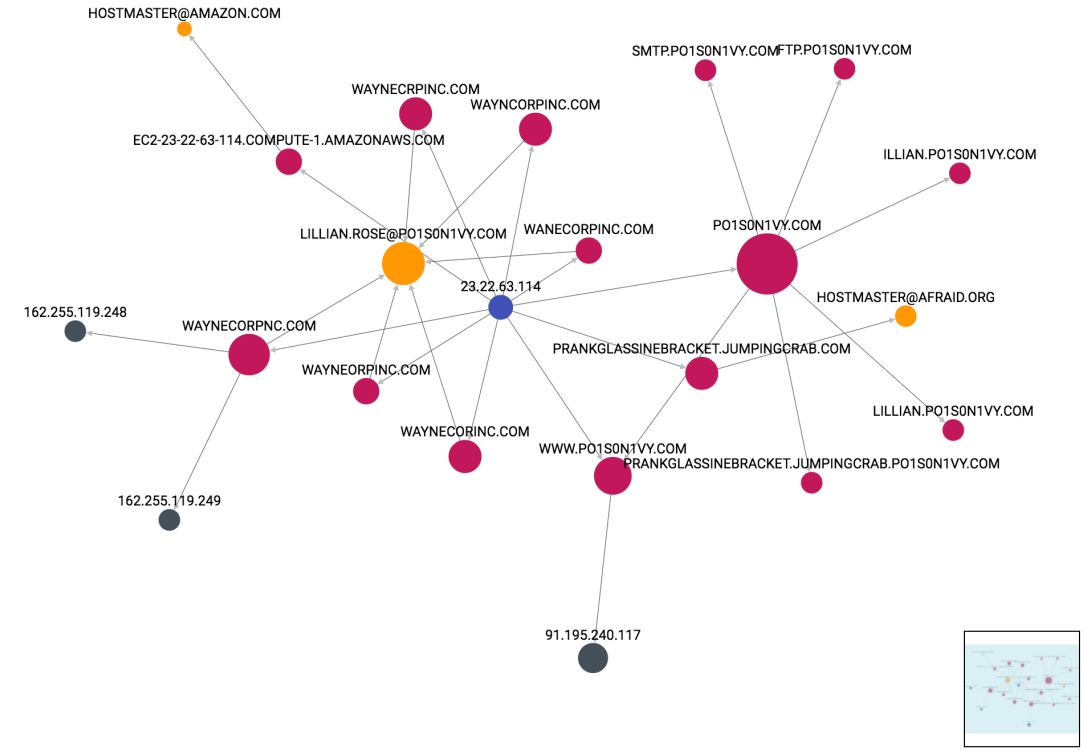
Hints

- Sites like VirusTotal and ThreatCrowd will help identify the domain to IP association

ThreatCrowd Domain Search



ThreatCrowd – Pivot to IP



IP WHOIS

Property	Value
Location	Ashburn, United States
Country	United States

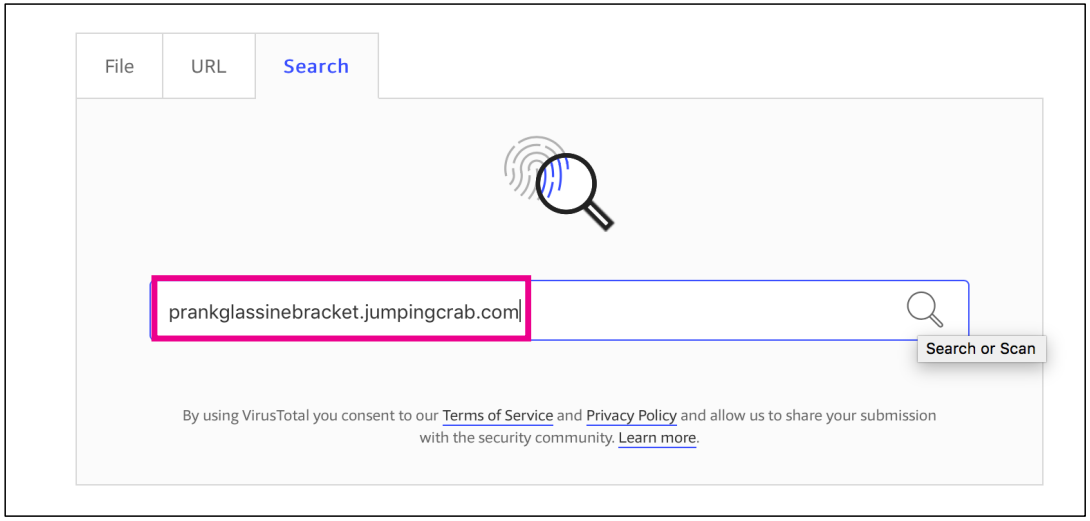
REVERSE DNS

Domain	Date
WYNECORPINC.COM	2018-11-04
wanecorpinc.com	2018-11-02
ec2-23-22-63-114.compute-1.amazonaws.com	2018-10-24
waynecorinc.com	2018-10-24
waynecrpinc.com	2018-10-24
wayneorpinc.com	2018-10-24
wayncorpinc.com	2018-10-23
waynecorpnc.com	2018-09-28
po1s0n1vy.com	2018-07-09
www.po1s0n1vy.com	2018-06-24
prankglassinebracket.jumpingcrab.com	2018-05-06

IP CLASSES

23.22.63..x=Browse , 23.22.63..x.x=Browse I View on ThreatMiner

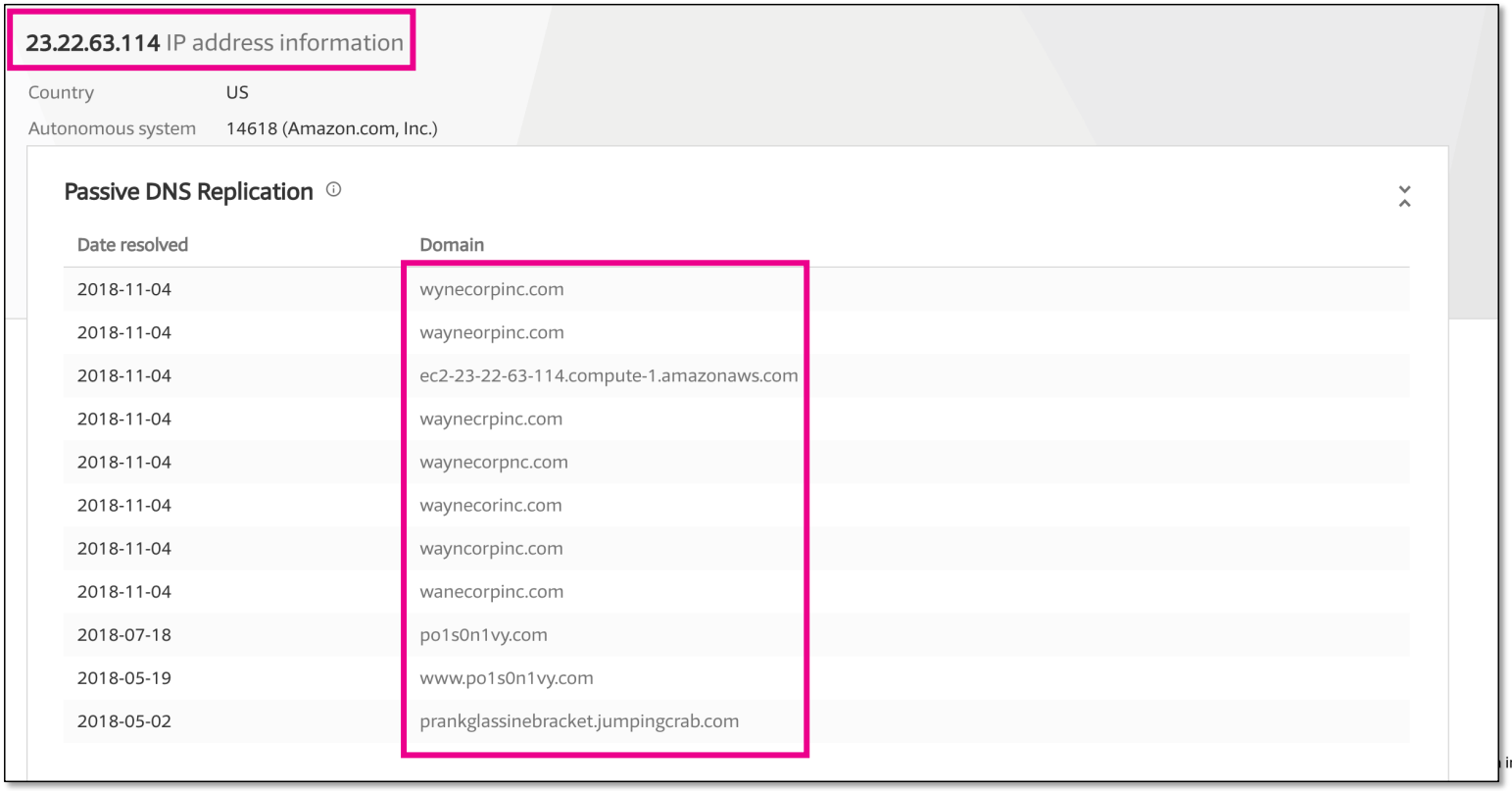
Virus Total



Passive DNS Replication

Date resolved	IP address
2016-09-14	23.22.63.114

VirusTotal



23.22.63.114 IP address information

Country	US
Autonomous system	14618 (Amazon.com, Inc.)

Passive DNS Replication

Date resolved	Domain
2018-11-04	wynecorpinc.com
2018-11-04	wayneorpinc.com
2018-11-04	ec2-23-22-63-114.compute-1.amazonaws.com
2018-11-04	waynecrpinc.com
2018-11-04	waynecorpnc.com
2018-11-04	waynecorinc.com
2018-11-04	wayncorpinc.com
2018-11-04	wanecorpinc.com
2018-07-18	po1s0n1vy.com
2018-05-19	www.po1s0n1vy.com
2018-05-02	prankglassinebracket.jumpingcrab.com

into doing

Using OSINT to Identify Attacker Infrastructure

Kill Chain Phase: Weaponization

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

- 23.22.63.114

Bonus Question: What additional domains were tied to that IP address?

- wynecorpinc.com
- wayneorpinc.com
- waynecrpinc.com
- waynecorpnc.com
- waynecorinc.com
- wayncorpinc.com
- wanecorpinc.com
- po1s0n1vy.com

Kill Chain

IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Executable Uploaded: 3791.exe
Hash of the Uploaded File: AAE3F5A29935E6ABCC2C2754D12A9AF0

File that Defaced Our Web Server: poisonivy-is-coming-for-you-batman.jpeg

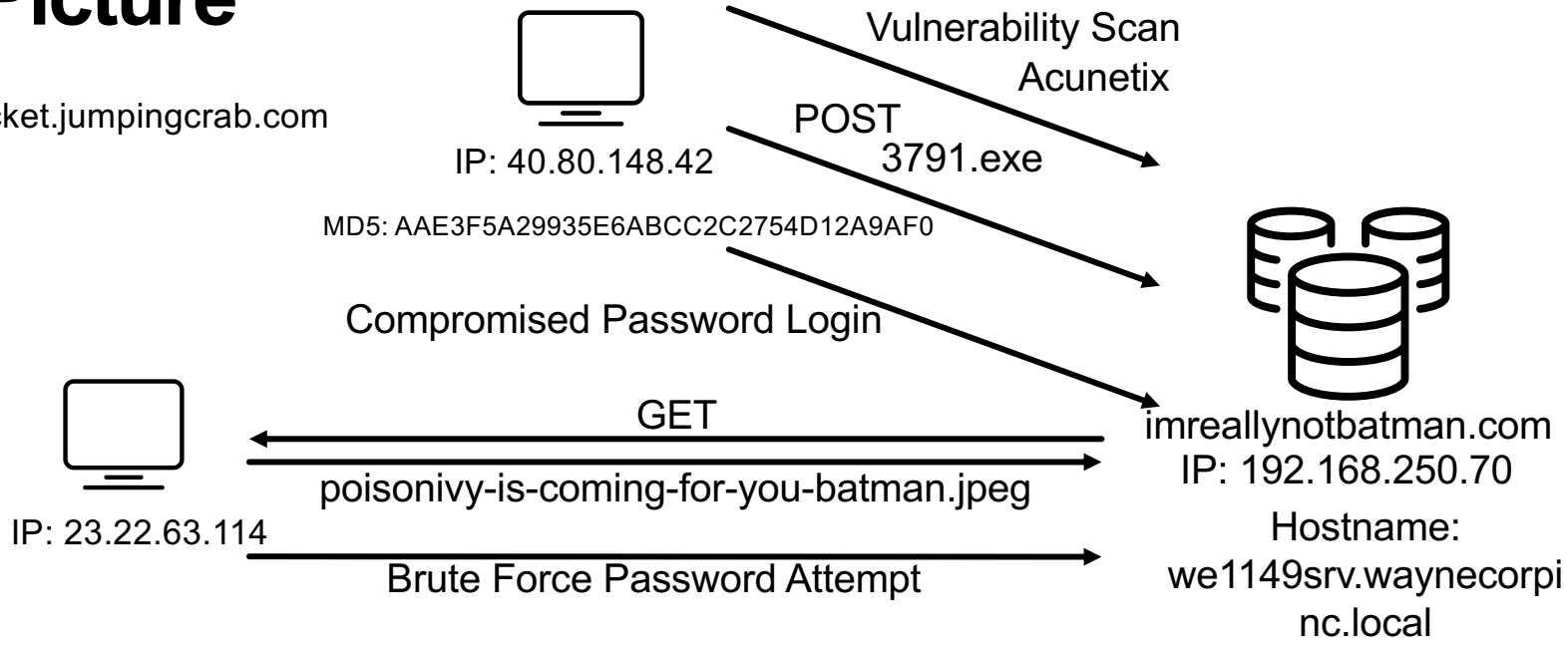
Attacker Infrastructure: wynecorpinc.com, wayncorpinc.com, waynecorinc.com, waynecrpinc.com, wayneorpinc.com, wanecorpinc.com, po1s0n1vy.com

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

Fully Qualified Domain Name of the System that Defaced the Web Server: prankglassinebracket.jumpingcrab.com

APT Picture

DNS:
prankglassinebracket.jumpingcrab.com
Other Domains:
wynecorpinc.com
wayncorpinc.com
waynecorinc.com
waynecrpinc.com,
wayneorpinc.com,
wanecorpinc.com
po1s0n1vy.com



Using OSINT to Identify Associated Malware

Kill Chain Phase: Delivery

GCPD reported that common TTPs (Tactics, Techniques, Procedures) for the P01s0n1vy APT group if initial compromise fails is to send a spear phishing email with custom malware attached to their intended target. This malware is usually connected to P01s0n1vy's initial attack infrastructure. Using research techniques, provide the SHA256 hash of this malware.

Bonus Question: What is the name of the file that was matches the hash?

Hints

- The initial compromise was successful, so the email was never sent. Splunk will not contain the answer.
- You have a number of indicators already collected, use these!
- ThreatMiner, VirusTotal and Hybrid-Analysis are all good OSINT sites to look for malware

Back to OSINT

What do we know about the attack infrastructure from our earlier investigation?

- IPs of Attack Infrastructure
- List of Domains

Anything else?

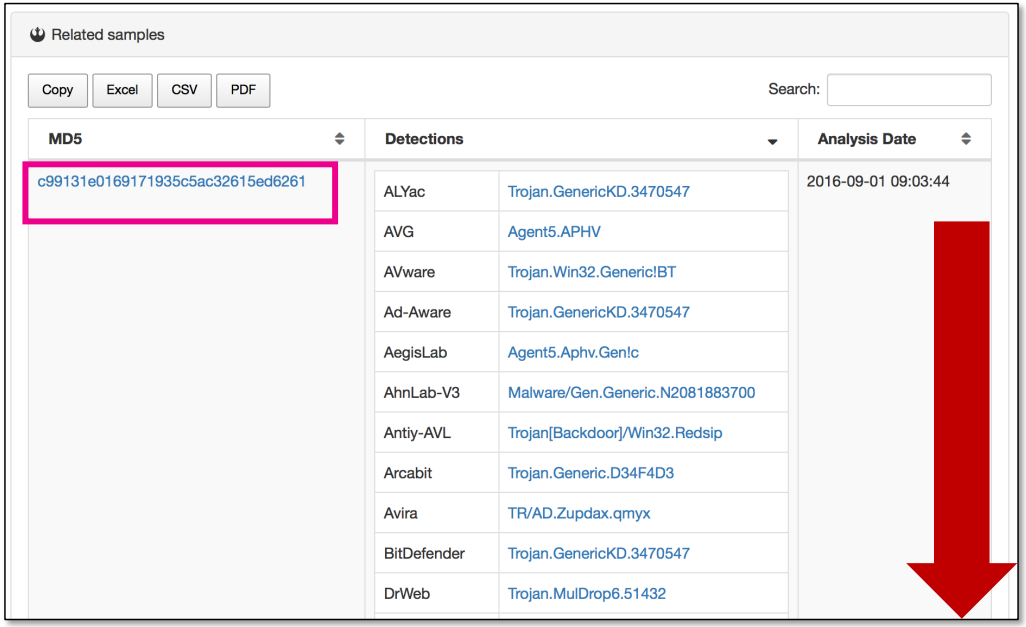
TheatMiner

Host: 23.22.63.114

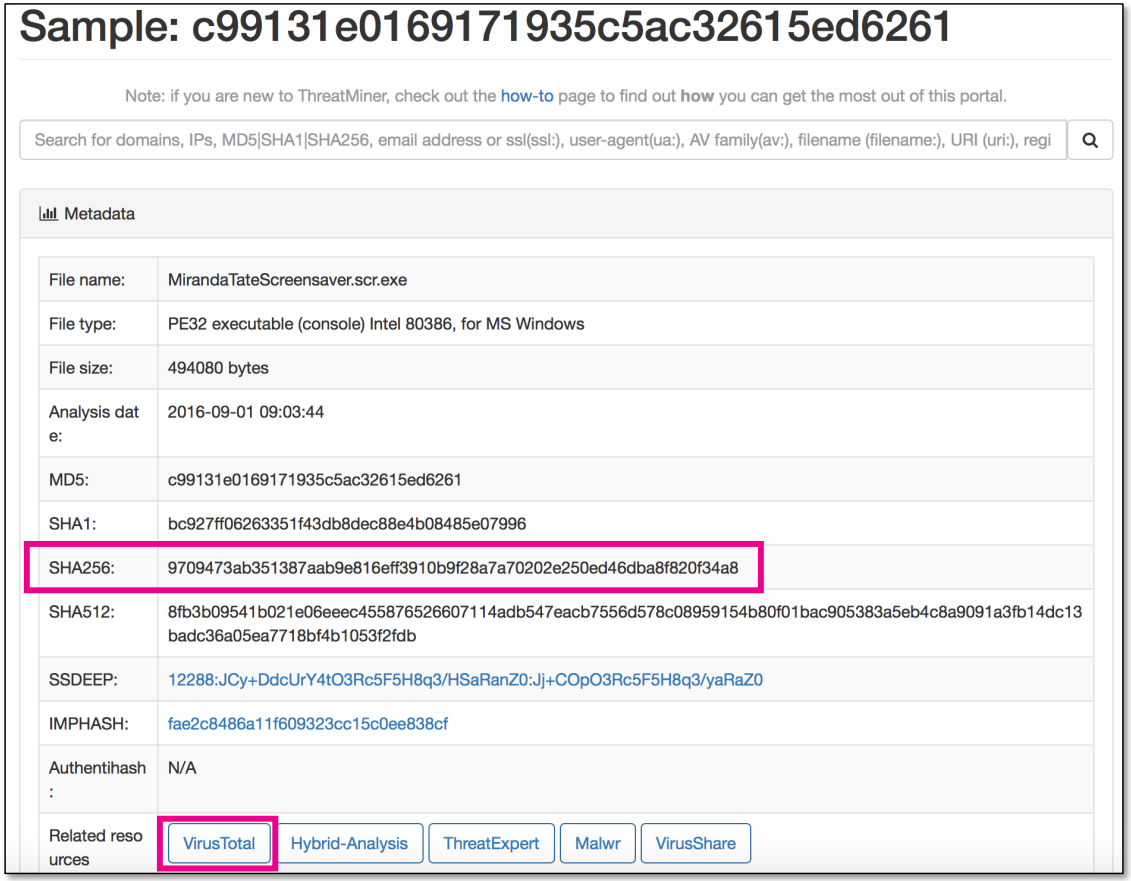
Note: if you are new to ThreatMiner, check out the how-to page to find out **how** you can get the most out of this portal.

Search for domains, IPs, MD5|SHA1|SHA256, email address or ssl(ssl:), user-agent(ua:), AV family(av:), filename (filename:), URI (uri:), regi

Threatminer can give me an MD5 of samples from 23.22.63.114 but not SHA256

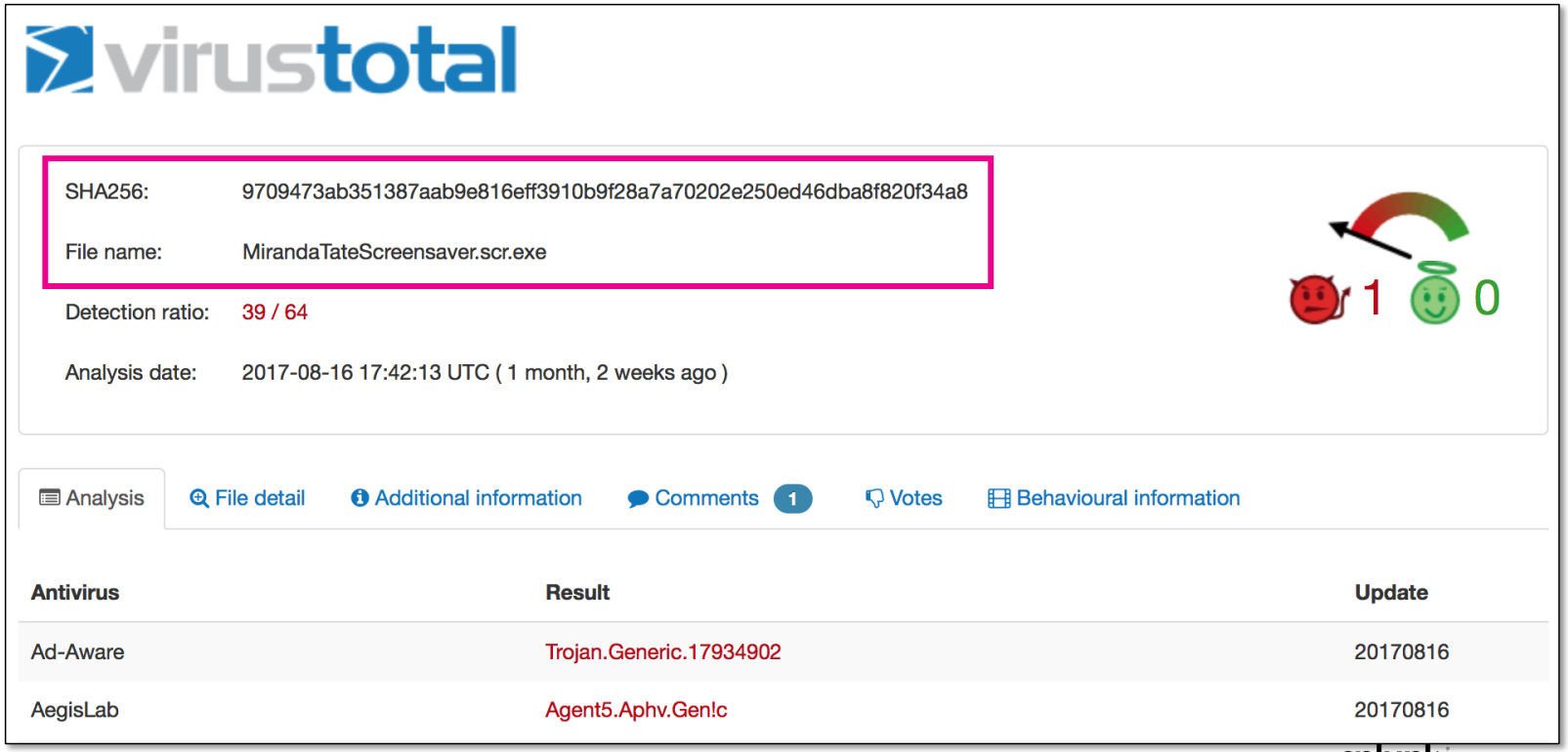


ThreatMiner - File Metadata



splunk> turn data into doing

VirusTotal - File Hash



Using OSINT to Identify Associated Malware

Kill Chain Phase: Delivery

GCPD reported that common TTPs (Tactics, Techniques, Procedures) for the Po1s0n1vy APT group, if initial compromise fails, is to send a spear phishing email with custom malware attached to their intended target. This malware is usually connected to Po1s0n1vy's initial attack infrastructure. Using research techniques, provide the SHA256 hash of this malware.

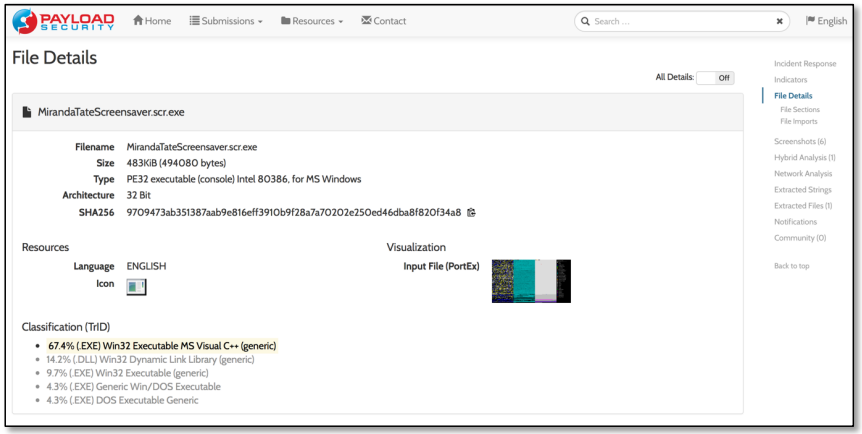
- 9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8

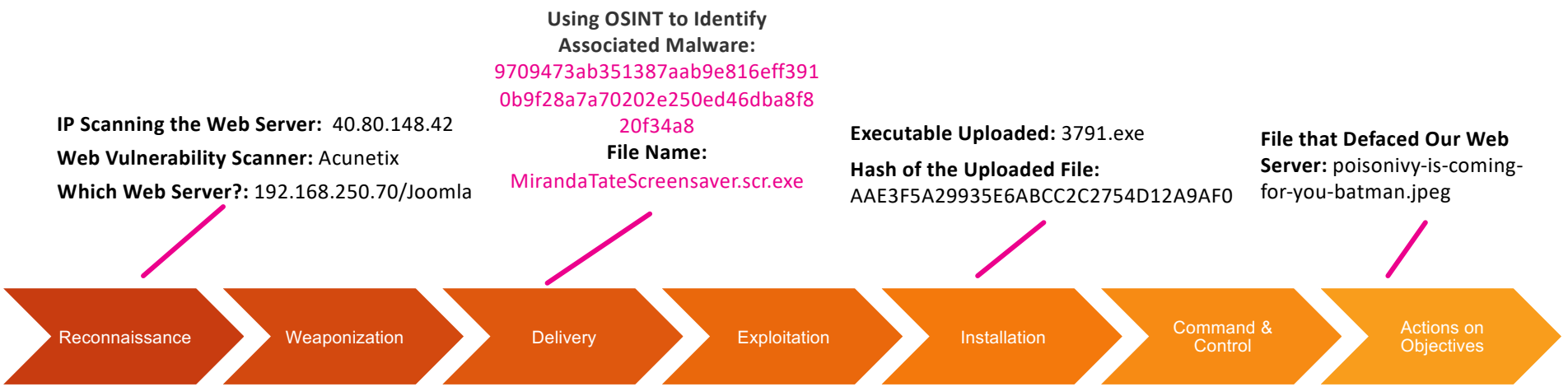
Bonus Question: What is the name of the file that was matches the hash?

- MirandaTateScreensaver.scr.exe

Other sites of interest:

- www.hybrid-analysis.com
- virusshare.com





IP Scanning the Web Server: 40.80.148.42
Web Vulnerability Scanner: Acunetix
Which Web Server?: 192.168.250.70/Joomla

Using OSINT to Identify Associated Malware:
9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8
File Name:
MirandaTateScreensaver.scr.exe

Executable Uploaded: 3791.exe
Hash of the Uploaded File:
AAE3F5A29935E6ABCC2C2754D12A9AF0

File that Defaced Our Web Server: poisonivy-is-coming-for-you-batman.jpeg

Attacker Infrastructure: wynecorpinc.com, wayncorpinc.com, waynecorinc.com, waynecrpinc.com, wayneorpinc.com, wanecorpinc.com, po1s0n1vy.com

Brute Force Attack Originated: 23.22.63.114
Identifying the First Password Attempted in a Brute Force Attack
Extracting Passwords from Events
Identifying the Password Used To Gain Access: 40.80.148.42
Determining The Elapsed Time Between Events

Fully Qualified Domain Name of the System that Defaced the Web Server:
prankglassinebracket.jumpingcrab.com

APT Picture

DNS:
prankglassinebracket.jumpingcrab.com
Other Domains:
wynecorpinc.com
wayncorpinc.com
waynecorinc.com
waynecrpinc.com,
wayneorpinc.com,
www.po1s0n1vy.com
wanecorpinc.com
po1s0n1vy.com

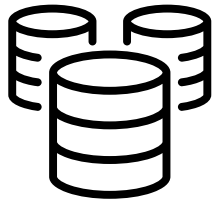
IP: 40.80.148.42

Vulnerability Scan
Acunetix

POST
3791.exe

MD5: AAE3F5A29935E6ABCC2C2754D12A9AF0

Compromised Password Login



imreallynotbatman.com
IP: 192.168.250.70

Hostname:
we1149srv.waynecorpinc.local

IP: 23.22.63.114

GET

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

Brute Force Password Attempt

Phishing Email

SHA256:
9709473ab351387aab9e816eff3910b9f28a7a70202e250ed46dba8f820f34a8
File Name: MirandaTateScreensaver.scr.exe

Domain:
waynecorpinc.local

splunk> turn data into doing

APT Summary

Scanned for vulnerabilities

Found site is running Joomla

Performed a brute force password scan, logged into Joomla, installed file upload modules

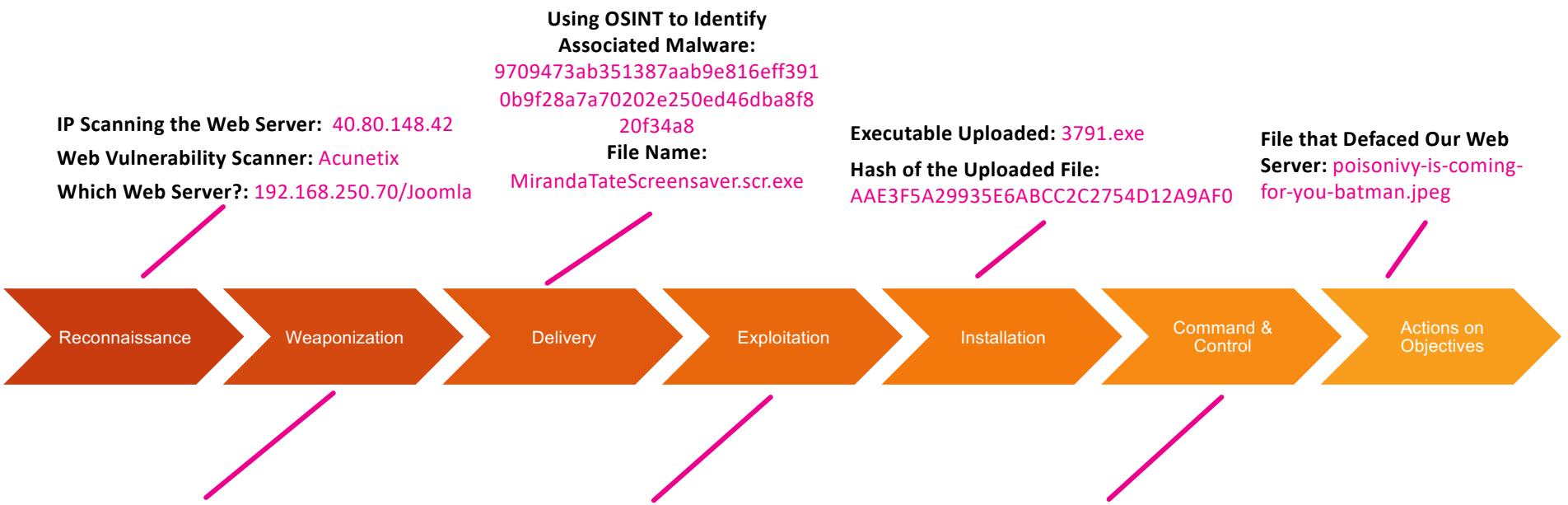
Uploaded webshell

Used webshell to upload reverse TCP shell

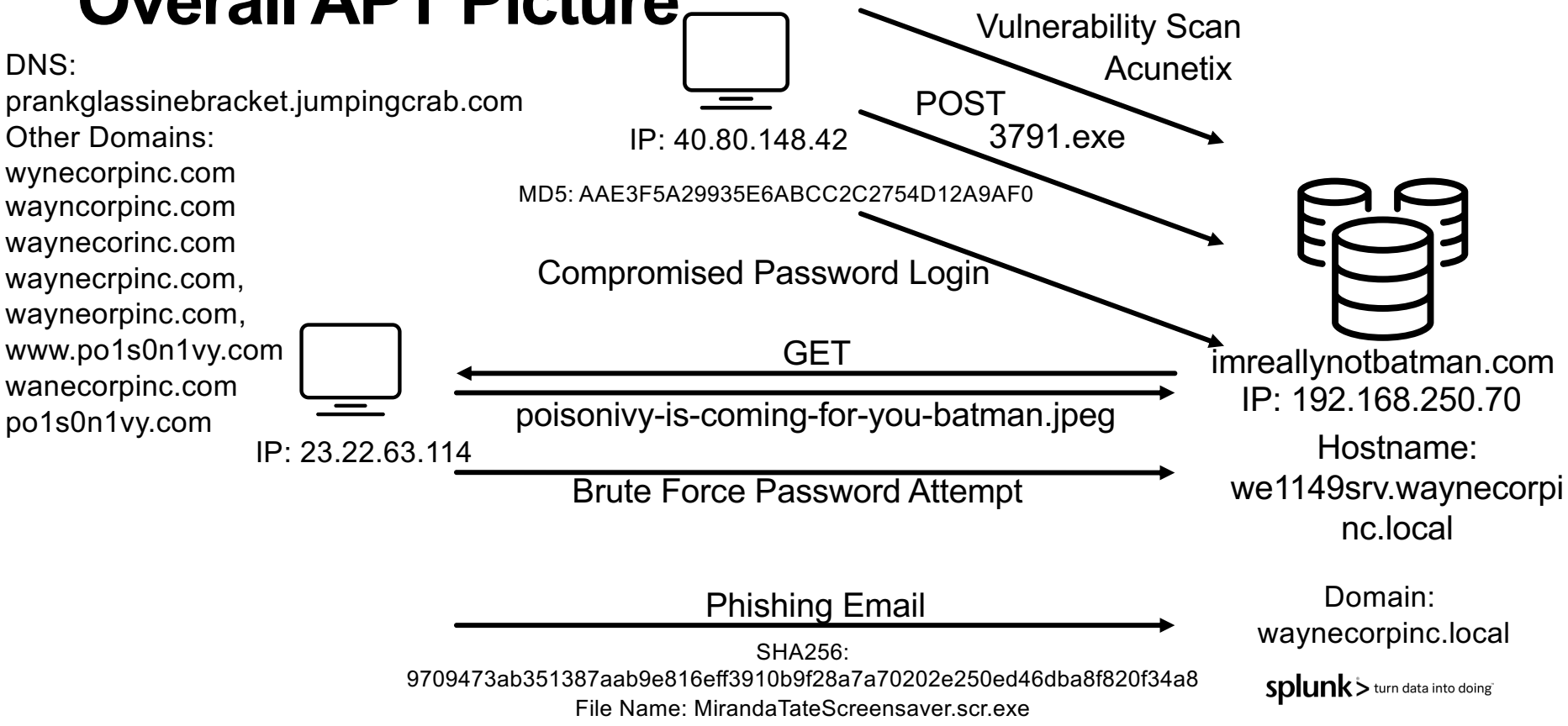
Connected via metasploit

Tried to move around but couldn't get out of locked down Windows 2012R2

Defaced website with downloaded defacement image



Overall APT Picture



Thank you!

splunk> turn data into doing™