

Protecting VSI From Future Attacks

Part 1: Windows Server Attack

```

a message 100+
a msad_action 3
a object 1
a object_category 2
a OpCode 1
a privilege_id 14
a Privileges 13
a Process_ID 100+
a Process_Name 7
a product 1
a punct 27
a raw 100+
# RecordNumber 100+
a Security_ID 100+
a session_id 15
# severity_id 2
a signature 15
# signature_id 100+
a SourceName 2
a splunk_server 1
a src_nt_domain 15
a src_nt_host 100+
a src_user 100+
a src_user_watchlist 1
a status 2
a subject 15
a ta_windows_action 1
a ta_windows_security_CategoryString 1
a tag 1
a tag::eventtype 1

```

Top 10 Values	Count	%
An attempt was made to reset an accounts password	4,256	35.771%
A user account was locked out	3,622	30.442%
An account was successfully logged on	864	7.262%
Domain Policy was changed	286	2.404%
The audit log was cleared	284	2.387%
A user account was changed	274	2.303%
A privileged service was called	272	2.286%
A process has exited	268	2.252%
A computer account was deleted	266	2.236%
A logon was attempted using explicit credentials	260	2.185%

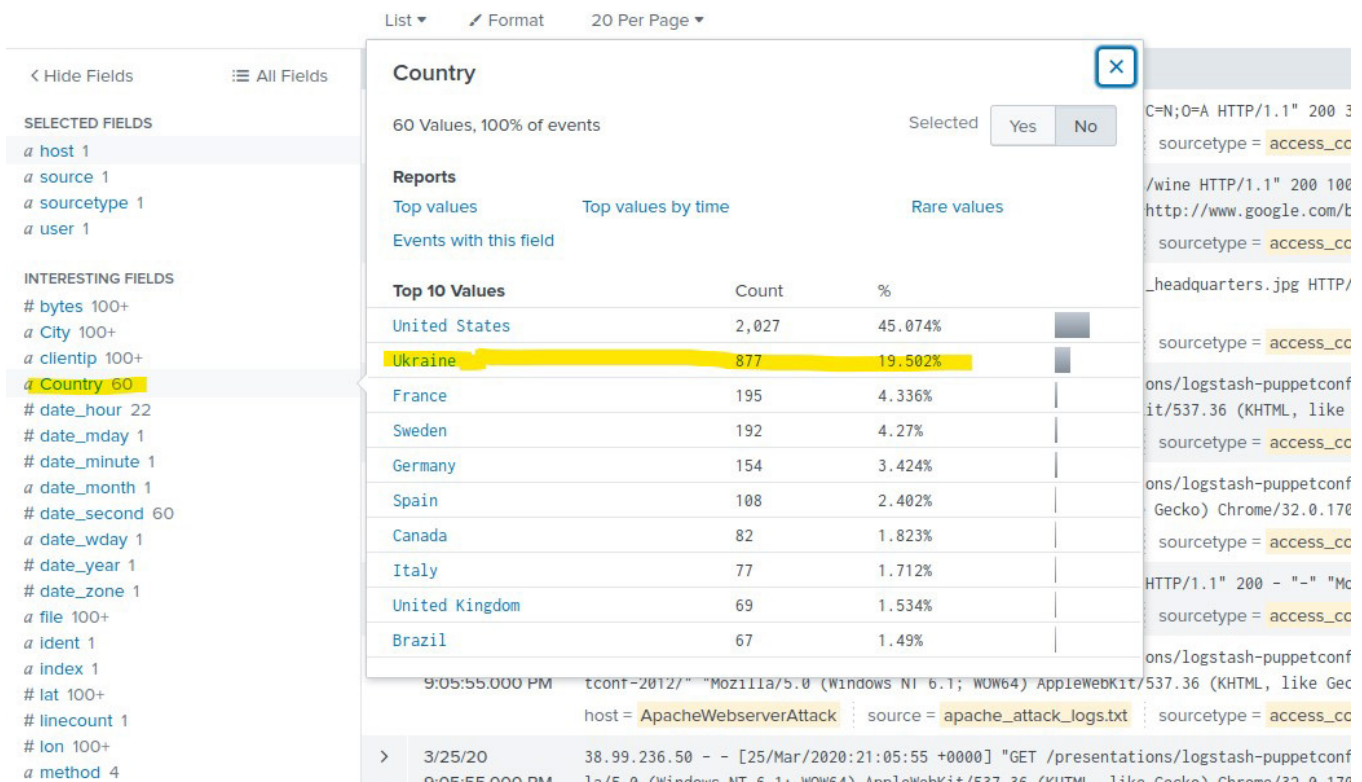
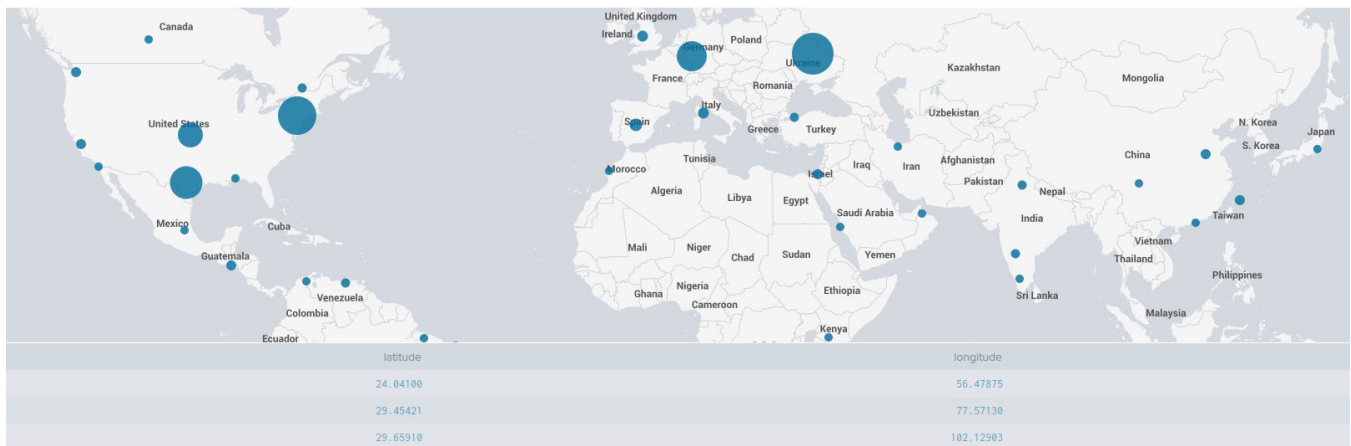
Question 1: One of the primary ways to prevent brute force attacks, or high number of account login attempts, with an end result of also locking out user accounts is to implement 2-factor/ multi-factor authentication as part of the login process. This will slow down the automated or brute force login actions that could cause lock outs to happen.

The other attack signature indicating a high level of account password reset attempts can be mitigated by prompting the user to manually type out the email address associated to the account in order to do a reset, and if the email is correct, a password reset link will be sent to the email on file. The confirmation page should be vague in confirming if the email is correct by showing the same message to the user whether the correct email was typed in or not.

Another thing to implement to the whole company as well is to require regularly scheduled password changes so that even if a password is compromised without knowledge, the duration that it is still applicable will be limited. Mandatory password changes can also be required to a specific user after such attacks occur and a successful login was made for a specific user.

Question 2: For VSI to protect against JobeCorp or another bad actor from trying to lock out every user, CAPTCHAs can also be implemented in addition to the above policies and mitigations so that locking any accounts is a highly time-consuming and laborious task. If anyone does try to persistently attempt any accounts to do a lock out, accounts should also be set to lock only for a set limited amount of time, for example 5-10 minutes, per lockout period before returning to normal, or incrementally time-out for longer and longer durations to not make sense to persist, yet allow the real user to login regularly at a later time.

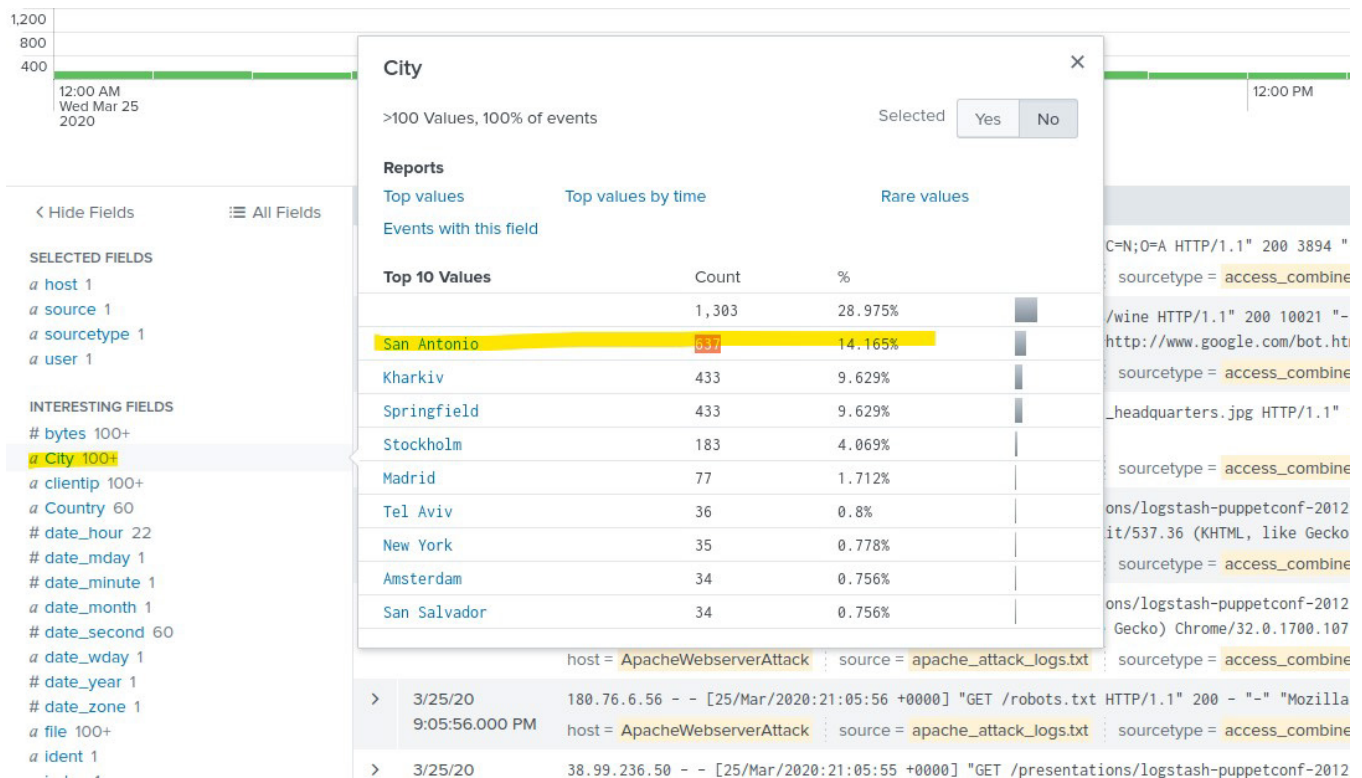
Part 2: Apache Webserver Attack



Question 1: One of the countries that has a high concentration of attacks originating is Ukraine. As a result, we will implement a firewall rule to block all traffic from them.

“Block all incoming HTTP traffic where the source IP comes from the country Ukraine”

Frank Lin - Unit 19 Homework - Protecting VSI From Future Attacks



There is also a lot of activity from specific cities where VSI does not do business, so these attacks are also safe to block.

“Block all incoming HTTP traffic from the cities San Antonio, Kharkiv, Springfield, and Stockholm”

Frank Lin - Unit 19 Homework - Protecting VSI From Future Attacks

INTERESTING FIELDS

- # bytes 100+
- a City 100+
- a clientip 100+
- a Country 60
- # date_hour 22
- # date_mday 1
- # date_minute 1
- a date_month 1
- # date_second 60
- a date_wday 1
- # date_year 1
- # date_zone 1
- a file 100+
- a ident 1
- a index 1
- # lat 100+
- # linecount 1
- # lon 100+
- a method 4
- a punct 100+
- a referer 100+
- a referer_domain 77**
- a Region 100+
- a req_time 100+
- a root 14
- a splunk_server 1
- # status 7
- # timeendpos 7
- # timesteppos 7
- a uri 100+

> 3/25/20 63.140.98.80 - - [25/Mar/2020:21:05:58 +0000] "GET /images/VSI_headquarters.jpg 36 (KHTML, like Gecko) Chrome/33.0.1750.91 Safari/537.36"

referer_domain

77 Values, 34.512% of events Selected

Reports

Top values Top values by time Rare values

Events with this field

Top 10 Values	Count	%
http://www.semicomplete.com	764	49.227%
http://semicomplete.com	572	36.856%
http://www.google.com	37	2.384%
https://www.google.com	25	1.611%
http://stackoverflow.com	15	0.966%
http://logstash.net	6	0.386%
http://tuxradar.com	6	0.386%
https://www.google.co.uk	6	0.386%
https://www.google.com.br	6	0.386%
http://kufli.blogspot.com	5	0.322%

Question 2: There is a high percentage of attack traffic coming from 2 variations of the same domain, www.semicomplete.com. There is a good chance that this webserver was compromised as a zombie computer or bot and being used in the brute force and denial of service attacks against VSI. About 85% of all attack traffic originates from this domain.

By blocking traffic coming from this referer domain, we should be able to also mitigate more attack traffic regardless of location to VSI's server.

```
Block all incoming HTTP traffic from the from the referer_domain 'www.semicomplete.com'
and 'semicomplete.com'"
```


Frank Lin - Unit 19 Homework - Protecting VSI From Future Attacks

< Hide Fields
All Fields

INTERESTING FIELDS

- # bytes 100+
- a City 100+
- a clientip 100+
- a Country 60
- # date_hour 22
- # date_mday 1
- # date_minute 1
- a date_month 1
- # date_second 60
- a date_wday 1
- # date_year 1
- # date_zone 1
- a file 100+
- a ident 1
- a index 1
- # lat 100+
- # lincount 1
- # lon 100+
- a method 4
- a punct 100+
- a referer 100+
- a referer_domain 77
- a Region 100+
- a req_time 100+
- a root 14
- a splunk_server 1
- # status 7
- # timeendpos 7
- # timestartpos 7
- a uri 100+
- a uri_path 100+
- a useragent 100+**
- a version 2

81 more fields
+ Extract New Fields

useragent

>100 Values, 99.978% of events

Selected Yes No

Reports

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

[Events with this field](#)

Top 10 Values	Count	%
Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.2; SV1; .NET CLR 2.0.50727987787; InfoPath.1)	1,296	28.826%
Chef Client/10.18.2 (ruby-1.9.3-p327; ohai-6.16.0; x86_64-linux; +http://opscode.com)	638	14.19%
Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107 Safari/537.36	291	6.472%
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.91 Safari/537.36	183	4.07%
UniversalFeedParser/4.2-pre-314-svn +http://feedparser.org/	84	1.868%
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:27.0) Gecko/20100101 Firefox/27.0	80	1.779%
Mozilla/5.0 (iPhone; CPU iPhone OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0 Mobile/10A5376e Safari/8536.25 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	74	1.646%
Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107 Safari/537.36	73	1.624%
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:27.0) Gecko/20100101 Firefox/27.0	72	1.601%
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107 Safari/537.36	69	1.535%

There is also a very high amount of the attack traffic coming from an older outdated version of the Mozilla/4.0 client. This seems to have a known association with a HTTP DDoS tool that is being used for attacks. Adding this to the firewall block rules would be beneficial for limiting the exposure to known attack tools.

```
"Block all incoming HTTP traffic from the from the user agent 'Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.2; SV1; .NET CLR 2.0.50727987787; InfoPath.1)'"
```