NETWORK SECURITY – Observing Attackers in the Wild

Project Report

Table of Contents

Contributions

Tasks	**Names**
SSH	Santosh Ghosh, Raghavendra Kumar
FTP	Sree Harsha, Venkata Balasubrahmanyam Dontu
POP3	Sree Harsha
Wordpress	Venkata Balasubrahmanyam Dontu
Joomla	Raghavendra kumar
Logging(Kibana, Elasticsearch, Logstash)	Santosh Ghosh
Analysis of the logs and answering the questions	Venkata Balasubrahmanyam Dontu, Raghavendra Kumar, Santosh Ghosh, Sree Harsha

1. Introduction

In this project, three honeypots are deployed on three sites – Japan, Ireland and USA. SSH, FTP and POP3 daemons are configured to log the attacker details.

2. Experimental Setup

For the purpose of analyzing the nature of the attacks being carried out on the honeypots, we install five well known services viz. SSH, POP3, FTP, Joomla, Wordpress on three servers at three locations (USA, Japan and Ireland). These services are modified to collect additional information than what they would normally do.

The setup for the experiment included a combination of the following tools:

1. Logstash Server : For collecting and parsing logs
2. Redis Server : For buffering remote log events (mainly a performance booster)
3. Elasticsearch : For indexing/storing the log events
4. Kibana : For querying and visualization of the results.
5. Modified deamons for SSH,POP3,FTP, Wordpress, Joomla

2.1 Architecture:

The overall architecture of the setup was as follows:

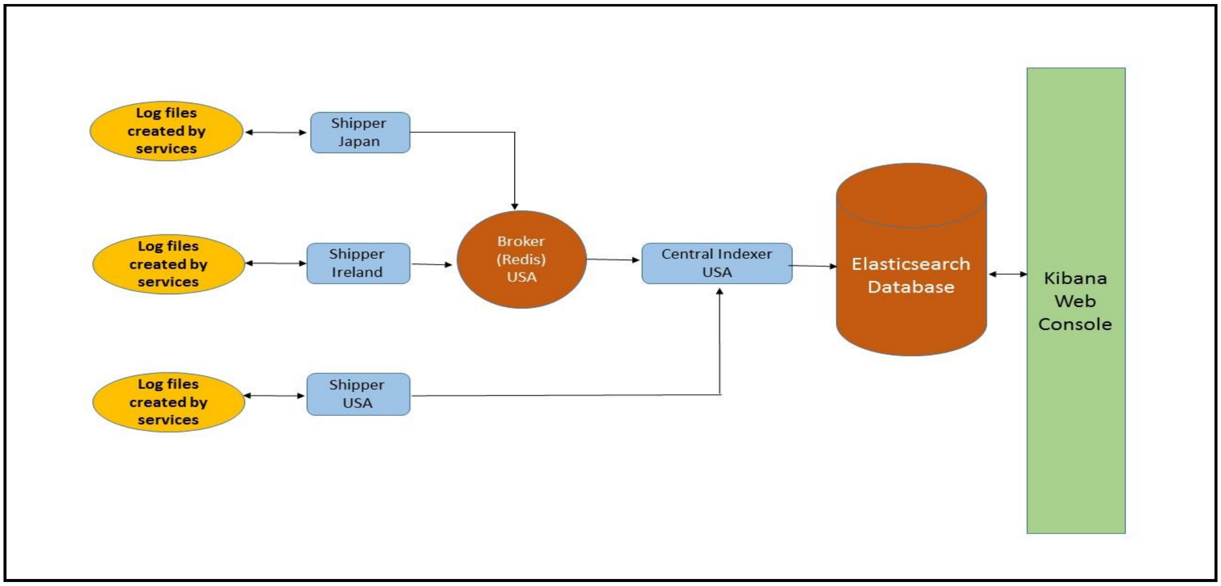

Figure 1: Architecture diagram of the setup

The ***Shippers*** are Logstash servers which collect the log files and send the data to the central Redis server located in USA.

The ***Broker*** in the diagram is the Redis server that is mainly used to buffer the remote logs till the central Logstash server parses them. It is mainly a performance choice so that the Central Indexer does not get overwhelmed by the constant stream of logs sent by the Shippers.

The Logstash server in USA is configured to serve a dual purpose. Like the other Shippers it collects its local logs but does not send them to Redis. It collects all the logs - its local logs and the logs collected from Redis, then parses all the logs (both local and remote)

The ***Central Indexer*** is the Logstash server installed in USA which is responsible for parsing the log events sent by the Shippers.

The ***Elasticsearch*** document database stores the log events as key value pairs. This was again a design consideration as the dominating action to be performed on the data is query not update. So a document database was the preferred choice.

Finally the ***Kibana*** web console is for querying and visualizing the stored data.

2.2 Event Life cycle:

The daemons and webservers are programmed to capture appropriate data and write them out to specific log files like syslog, mail.log and auth.log etc. Each such write to the logfiles is treated as an event by the logstash server, which then picks up the event and sends it to the central logstash server. The central logstash server then parses the logs as per the configuration scripts and stores the captured data in the Elasticsearch database server. The Kibana console interfaces with the Elasticsearch database and is used for querying the database and visualizing the results.

3. Configuring the daemons:

Each of the daemons deployed on the servers were programmed to log information like username, password, IP address of the attacker and timestamp of the attack. The important modifications are mentioned below. There is a step-by-step configuration and implementation details are submitted in a separate file (see README).

3.1 SSH Configuration: Version OPENSSH 6.6.1p1

File modified: **auth2-passwd.c**
Relevant parts of code modified: (Code at line 75 of auth2-passwd.c)

```
if(authenticated == 1){
        logit("CSE523 Succesful Authentication attempt using password: %s for %s from %s port
%d",password,authctxt->user,get_remote_ipaddr(),get_remote_port());
    }
    else if(authenticated == 0){
    logit("CSE523 Failed Authentication attempt using password: %s for %s from %s port
%d",password,authctxt->user,get_remote_ipaddr(),get_remote_port());
    }
```

Log message format:

Apr 19 06:51:01 ip-172-31-37-7 sshd[22122]: CSE523 Failed Authentication attempt using password: ketome for root from 43.255.190.92 port 43455 [preauth]

3.2 Dovecot Configuration: Version dovecot-pop3d version 2.2.9

Files edited: Files in **/etc/dovecot/dovecot/conf.d** folder

- 10-ssl.conf (To avoid configuring SSL for dovecot authentication)
- 10-auth.conf (Enabling authentication with plain text)
- 10-logging.conf (logs username and passwords tried in the format mentioned below)
- 10-mail.conf(configure inbox for successfully logged in users)

Log Message Format:

Apr 1 04:47:21 ip-172-31-18-190 dovecot: auth-worker(28822): pam(apr1,127.0.0.1): pam_authenticate() failed: Authentication failure (password mismatch?) (given password: pass_april)

3.3 FTP Configuration: Version pure-ftpd-1.0.36

File edited: *ftpd.c*

```
  //On unsuccessful Login attempt.
  if (authresult.auth_ok != 1) {
    logfile(LOG_INFO, "CSE508_FTP Accessed by UserName: %s and Password: %s and Auth: Failed", account,
pwd);
  }
  //On successful login attempt.
  else
  {
    logfile(LOG_INFO, "CSE508_FTP Accessed by UserName: %s and Password: %s and Auth: Success", account,
pwd);
  }
```

Log Message Format:

```
Apr 27 00:34:44 ip-172-31-37-7 pure-ftpd: (?@de222-089.resnet.stonybrook.edu) [INFO]
CSE508_FTP Accessed by UserName: admin and Password: root and Auth: Failed
```

3.4 Joomla Configuration:

Installation Steps Followed: https://www.howtoforge.com/how-to-install-joomla-on-ubuntu-14.04
Edited file: */var/www/html/joomla/plugins/authentication/joomla/joomla.php*

Added the snippet of code at line 72 to manipulate log file. The exact code is mentioned in the separate file “Detailed_Installation.docx” (see README).
To enable comment box. Downloaded and installed jcomment plugin

Log message format:

```
Mar 30 16:26:58 localhost apache2: CSE508_joomla unsuccessful login
from     IP:172.24.16.239 user:sddcs password:dsccs
```

3.5 Wordpress Configuration:

File edited: /var/www/html/wp-login.php

Relevant lines edited: Line number 748 of the above file

To allow automatic comments, we removed email and username as mandatory fields.

Log Message Format:

```
Apr 27 01:03:07 ip-172-31-37-7 apache2: CSE508_wordpress Accessed with UserName: admin and
Password: root123 from IP:130.245.222.89
```

3.6 Logstash Configuration:

We used regular expression script running on Logstash to parse out information form the log messages mentioned above. The detailed Logstash scripts are included in the XXXXXXX folder.

4. Results

4.1 Distribution of authentication attempts per service per server

Site	SSH	POP3	FTP	Joomla	Wordpress
Ireland	440406	216	6	15	9
Japan	1126090	32410	21	9	6
USA	919277	136	10	12	6
Total	2485773	32762	37	36	21

Table: 1 Distribution of attacks on servers across services

4.2 Service wise authentication attempts:

From the above statistics collected as of April 25, SSH attacks are very high compared to POP3 attacks. FTP, Joomla and Wordpress did not register any attacks. Large number of attacks on SSH makes sense because it gives complete control of victim machine. So the number of attempts to break in are high.

4.3 Distribution of distinct IP addresses attacking per service:

Service	Unique IPs Count
SSH	2783
POP3	11
FTP	8
Joomla	4*
Wordpress	4*

Table: 2 Unique IPs per Service

* All four IPs used by us for testing purposes

4.4 Empirical Cumulative Distribution Functions

4.4.1 ECDF of IP Addresses:

The empirical CDF for the IP, User Names and Passwords gives us an idea of the distribution of these entities.

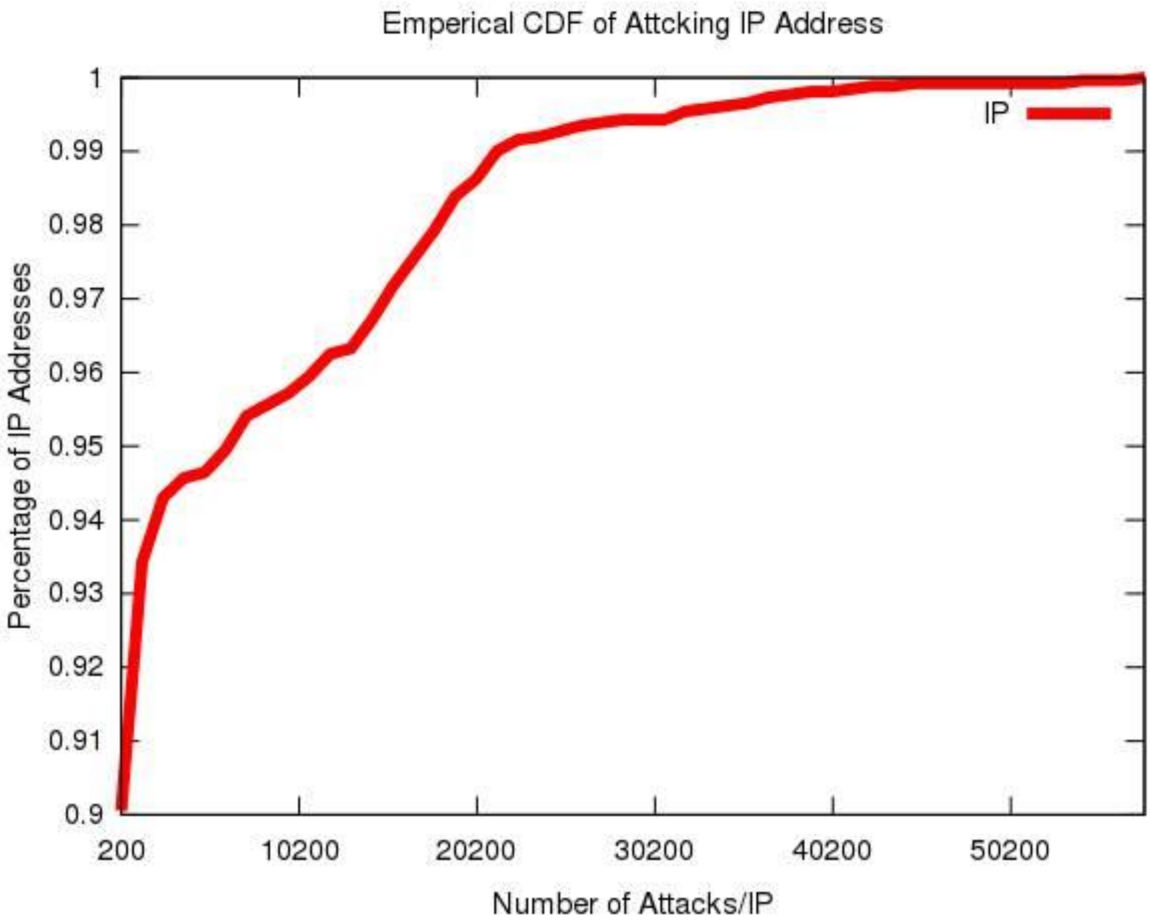

Figure: 2 ECDF of IP

From the CDF of IP address we see that 90% of the IP addresses logged were involved in doing 200 (0.33%) or less attacks. But only about 5% of the IP addresses did 10,000 or more attacks. This shows that the majority of the attacks to break-in were done by a small percentage of the IP addresses. To be precise 90% of the IPs did about 200 attacks while the only 1% of the IPs did 30,000 or more attacks.

4.4.2 ECDF of Passwords:

Coming to the CDF of the passwords we see a similar trend. About 99.3% of passwords were used to break into services 100 or less times. But the CDF takes a very steep curve at around the 500 mark indicating that about 99.8% passwords were used to break-in 500 or less times. From this we can say that only a small percentage of the passwords were used heavily to brute force the services.

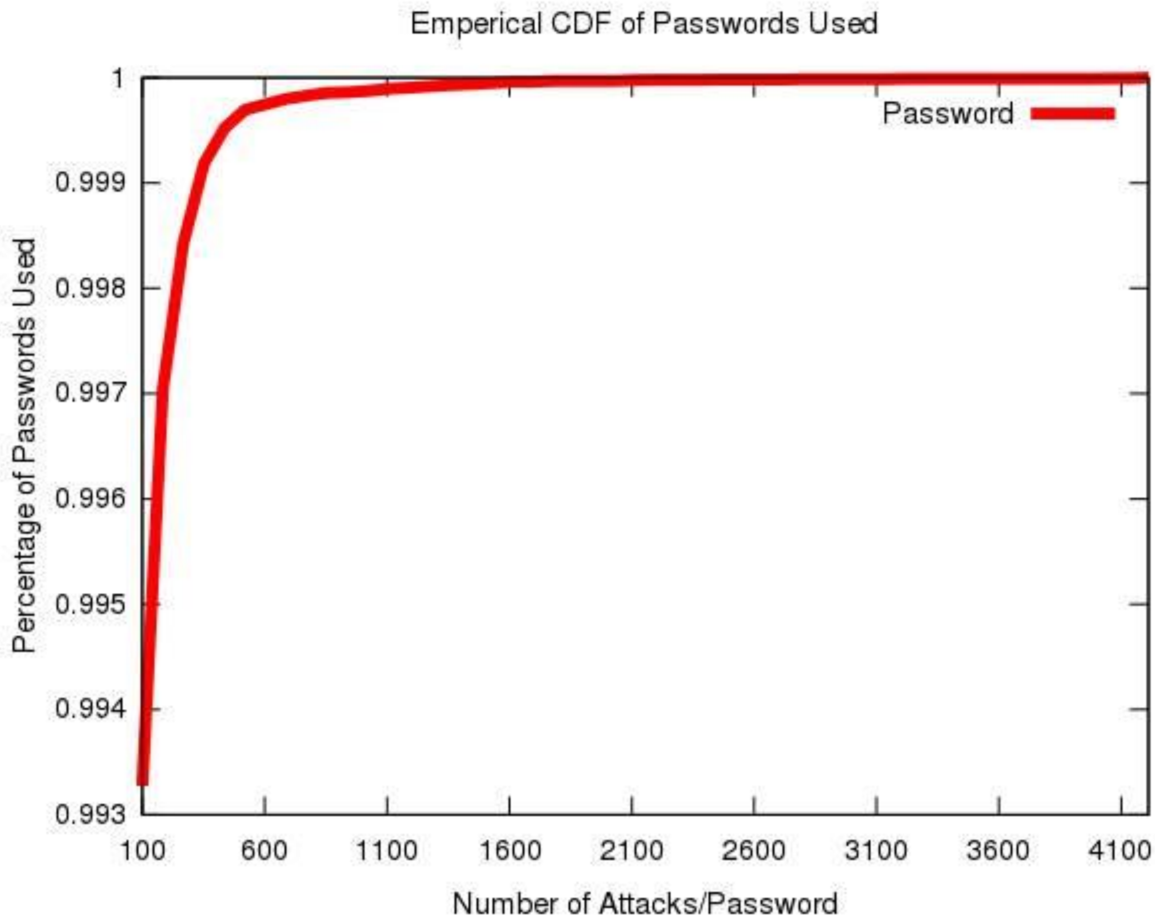

Figure: 3 ECDF of Passwords

4.4.3 ECDF of Usernames:

The CDF of usernames shows something interesting. The most used username was "root" which was used in almost 99.9% of the attacks. This explains the near perpendicular nature of the username CDF.

4.5 Blacklisted IPs

There are total of 2529 unique IP addresses on all Servers. Out of which 161 match with blacklisted IPs. The percentage of IP addresses is 6.36%.

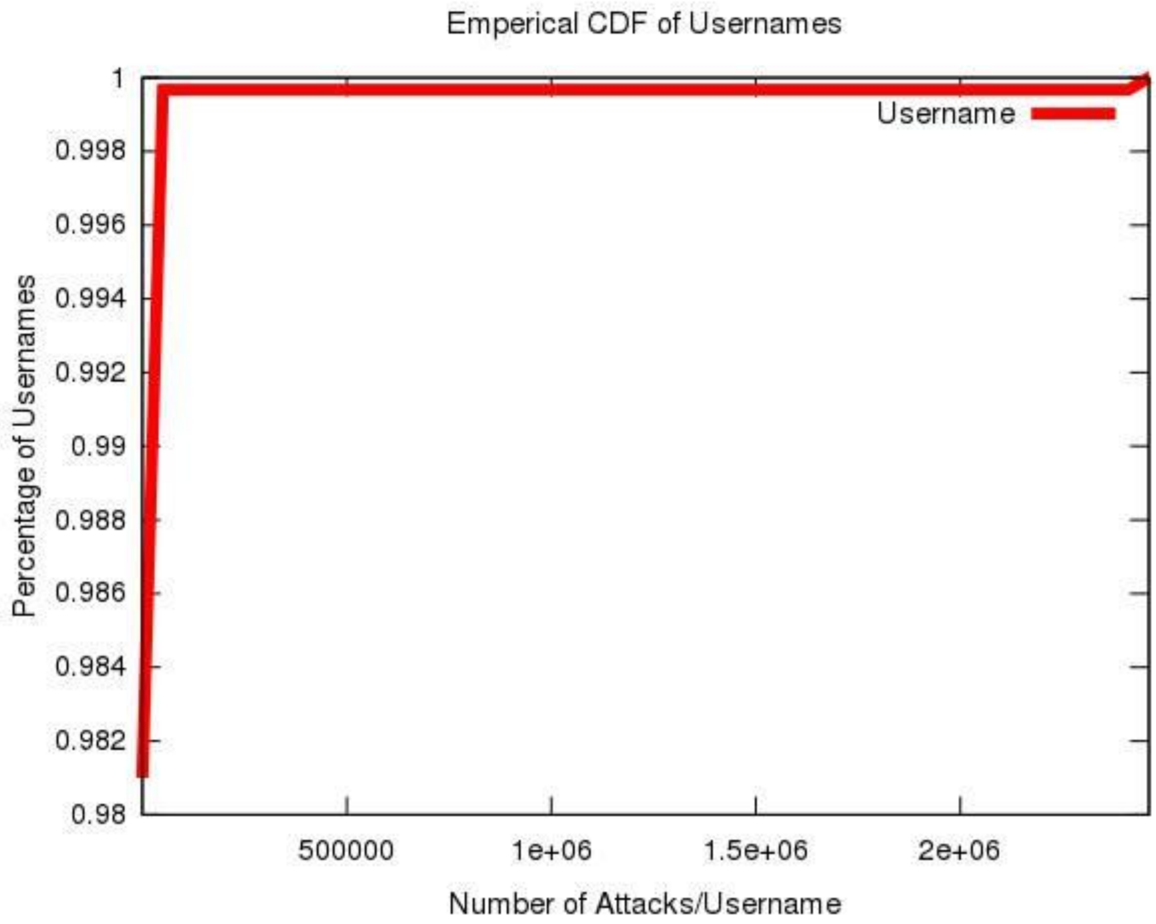

Figure 4: ECDF of Usernames

4.6 Distribution of top ten usernames:

Usernames	Count
root	2511952
admin	3653
test	2764
guest	1350
nagios	1303
zabbix	835
oracle	803
user	769
ubnt	579
ubuntu	557

Table 3: Top 10 User Names

4.7 Distribution of most Common Passwords

Passwords	Count
123456	4286
wubao	4248
jiammia	4066
password	3212
12345	2824
test	2459
1234	2107
abc123	2105
admin	2094
1234567	1668

Table 4: Top 10 Passwords

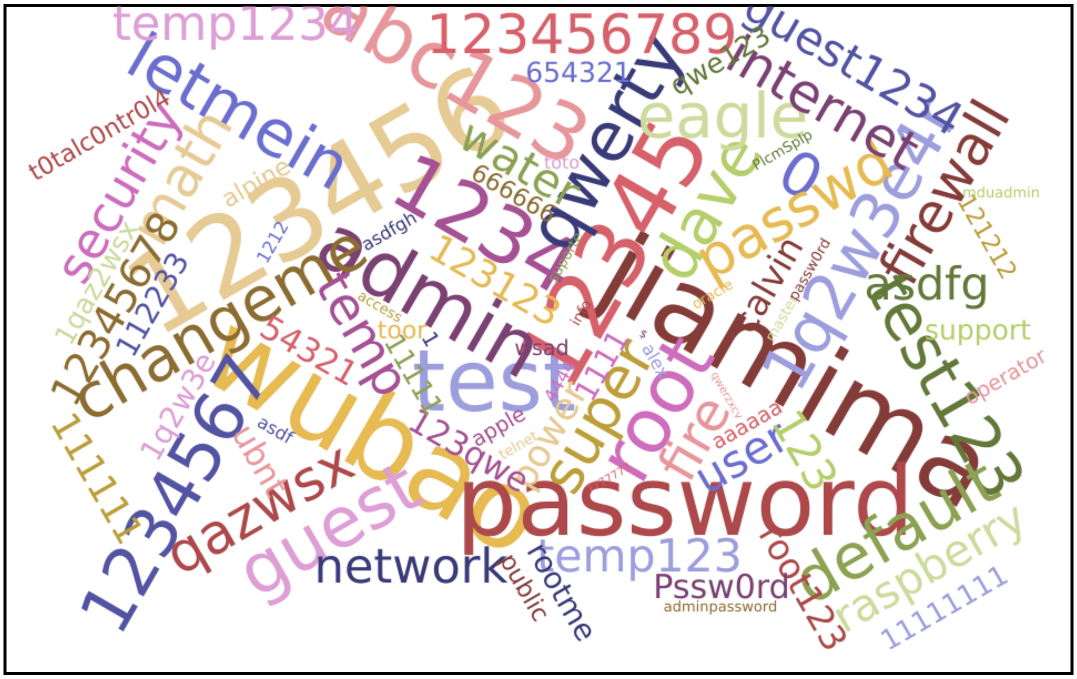

Figure 5: Word Cloud showing top 100 passwords

4.8 Temporal Trends in attacks:

4.8.1 Hourly

We defined night as 8pm-8am and day as 8am-8pm. All times are local to servers where services are deployed. The percentage of attacks during night time are 62.94% where as 37.7% of attacks happened during the day.

4.8.2 Weekly:

We defined Monday-Thursday as Weekdays and Friday-Sunday as Weekends.
It is observed that 17.73% of attacks happened on weekends while 82.27% of attacks occurred on weekdays.

4.9 Observations on different sites regarding the nature of attacks:

From the data collected it is observed that the number of attacks on Japan server was more than that of Ireland (more than double) but the number of IPs that attacked each server was almost equal. Also we find that among the IPs attacking these two servers 16% is common.

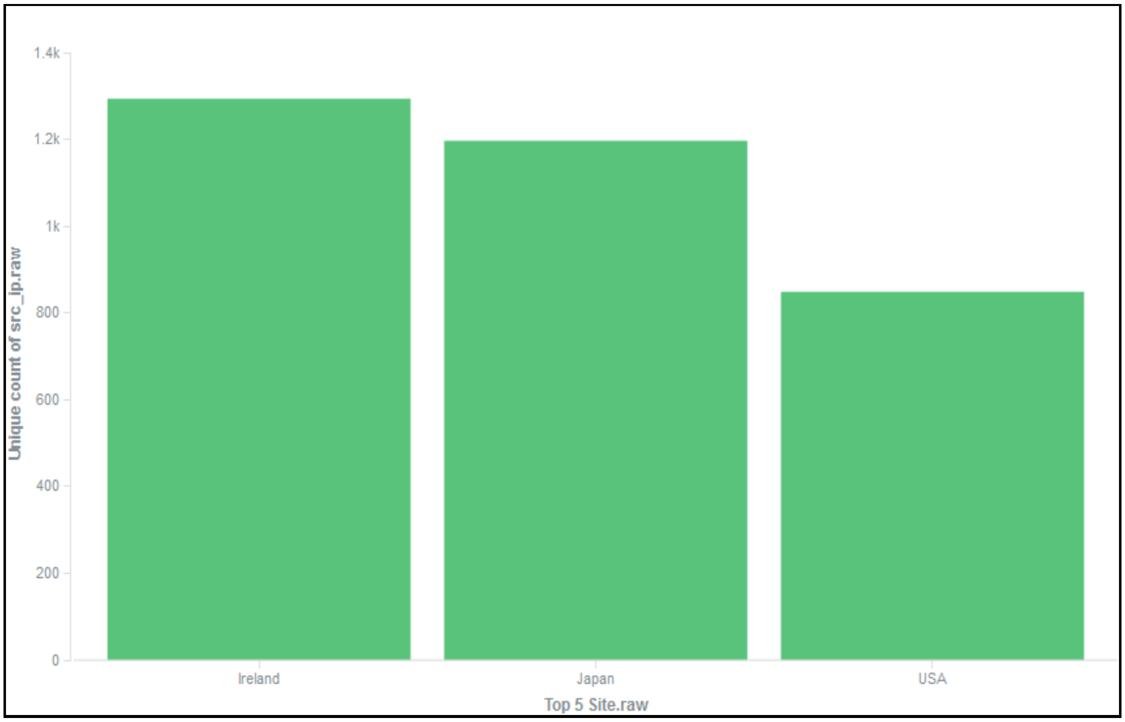

Figure 6: Unique attacking IPs per site

4.9.1 Common attackers (IP addresses) between Servers and Services:

While analyzing the common IPs across servers we find that about 6% of the all the IPs logged attacked all three servers. Also a whois lookup of the common attacking IPs show that almost 50% of them were registered under APNIC from HongKong and belonged to a common block namely 43.255.190.1/24.

There are no common attackers across services. Although there were few common authentication attempts they were primarily done by us for testing purposes.

Common IPs Across	Number of Common IPs	Percentage
USA & Japan	241	11.5%
USA & Ireland	291	19%
Japan & Ireland	319	16%
USA & Japan & Ireland	199	6.1%

4.10 Spam Comments on Wordpress and Joomla Websites:

We did not register any comments/login attempts on the wordpress and joomla services.

4.11 Country-wise profile of Attackers:

We analyse the location of the IP addresses that attacked the services by looking up the IP addresses in the most recent version on the publicly available Maxmind database (https://www.maxmind.com/en/geoip2-databases) . An overwhelming percentage of the attacks originated from Hong Kong followed by China.

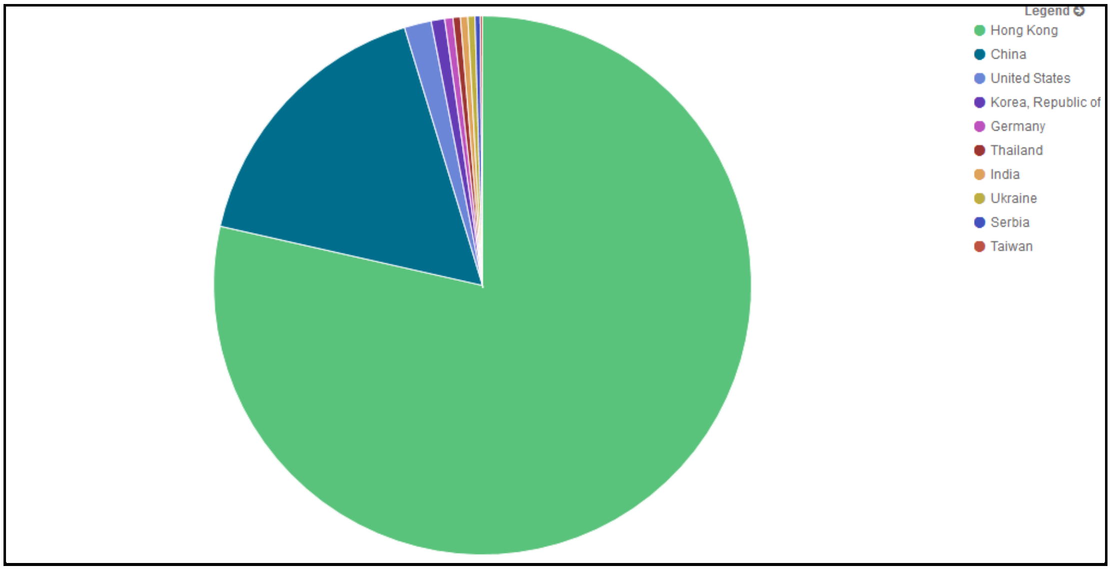

Figure 7: Country wise breakup of attackers

5. Conclusion:

We have given a detailed analysis of attacks on the USA, Japan and Ireland servers across SSH, FTP, Joomla, Wordpress and POP3 services over a period of 30 days.