DEGREE PROJECT IN COMPUTER ENGINEERING,
FIRST CYCLE, 15 CREDITS
STOCKHOLM, SWEDEN 2019

Implementing security techniques to lower the probability of IoT-devices getting hacked

ANDREAS WALLSTRÖM

MOHAMMAD-ALI OMER

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Implementing Security Techniques to Lower the Probability of IoT-devices Getting Hacked

ANDREAS WALLSTRÖM AND MOHAMMAD-ALI OMER

Master in Computer Science
Date: May 31, 2019
Supervisor: Robert Lagerström (KTH) and Bruce Edward DeBruhl (Cal Poly)
Examiner: Örjan Ekeberg
School of Electrical Engineering and Computer Science
Swedish title: Minska sannolikheten för IoT enheter att bli hackade

Abstract

IoT security is something that is becoming more important with the exponential growing number of IoT devices. It is important to find methods that can make IoT devices more secure and are feasible to install and use. This paper investigates how effective the security features geographical IP based blocking (GeoIP) and a limit on the number of allowed sign-in attempts to a server (fail2ban) are at reducing the number of successful hacker attacks. By launching honeypots with and without these security features data was collected about the number of hacking attempts. The results shows that the GeoIP security feature can reduce attacks by roughly 93% and that fail2ban can reduce the attacks by 99%. Further work in this field is encouraged to create better GeoIP tools and to better understand the potential for these security techniques on a larger scale.

Sammanfattning

IoT-säkerhet är ett fält med en allt mer ökad relevans i dagens samhälle i och med den exponentialla tillväxten av IoT-enheter. Det är viktigt att hitta metoder som kan göra IoT-enheter säkrare och är enkla att installera och använda. Den här rapporten undersöker hur effektiva geografiskt baserad IP-blockningar (GeoIP) och en begränsning i antalet tillåtna inloggningsförsök till en server (fail2ban) kan vara i att minska antalet lyckade attacker mot IoT-enheter. Genom att sätta upp honeypots med och utan de tidigarenämnda säkerhetsfunktionerna kunda vi samla data på hur de påverkade antalet attacker. Resultaten visade att GeoIP reducerade antalet med ungefär 93% och att fail2ban reducerade antalet med ungefär 99%. Framtida arbete inom detta fält kan vara att skapa en snabbare och mer simpel GeoIP modul och att försöka förstå hur dessa säkerhetstekniker kan påverka IoT-enheter i en större skala.

Contents

Chapter 1

Introduction

Internet of Things (IoT) is an exploding field. Mozilla estimates that the number of IoT devices will be 30 billion in 2020 [1]. An IoT device is a physical device connected to the Internet that collects and gives information and can oftentimes execute a physical action on the information it gets.

IoT devices will have an integral part in the future of our everyday lives, for example in the development of Smart Cities and Smart Homes to name a few. The idea of a Smart City is a city with IoT devices that collects and gives information about city infrastructure, services and operations for example in hopes that this will efficiently optimize and better the work of city officials. Smart Homes are homes equipped with IoT devices that can control things such as lighting, temperature, IP cameras and even door locks. Since the IoT devices can be placed in critical situations in Smart Cities and Smart Homes where they can do a lot of harm if hacked, IoT security needs to be prioritized.

New security vulnerabilities are found all the time. In 2018 almost 1400 new security holes were reported every month[1] accessed 2019-02-27. In 2016 one of the world's largest distributed denial-of-service (DDoS) attack were executed from an IoT botnet [2]. With the increase of different IoT devices this number will probably continue to increase. With so many internet connected devices controlling systems and collecting data, security becomes a more than ever important aspect to focus on.

The most common way to attack IoT devices are with brute force attacks. 87 % of all hacked IoT devices are hacked by getting either their Telnet or SSH

[1] Source: https://www.cvedetails.com/browse-by-date.php

password guessed. Furthermore, 60 % of all attacks are originating from just 5 countries (Brazil, China, Japan, Russia and United States)[2]. Open Web Application Security Project (OWASP) is a foundation that among other things creates a report listing the top 10 most important security problems with IoT devices. OWASP states in their 2018 report that the top three security issues with IoT devices are:[3]

1. Using default credentials or easy to brute force credentials.
2. Having unnecessary or insecure network services running on the device, such as Telnet and SSH.
3. Insecure software running on the device that can be exploited.

Point 1 and 2 are interlinked. Hackers often get access to a device by making a SSH connection using weak credentials. Therefore it is clear that one of the most common ways to attack IoT devices is by getting access to the device through Telnet or SSH. Therefore, it is very relevant and important to focus on ways to protect and secure against these types of attacks.

One way to to protect against brute force attacks are by using the fail2ban software. To protect against attackers from other countries one can use a security technique called GeoIP ban. A detailed explanation of both fail2ban and GeoIP will follow in the background section.

1.1 Research Question

Since a vast majority of attacks are performed by brute forcing the credentials and most of all attacks are originating from five countries[4] this paper will focus on implementing security mechanisms that address these issues.

The questions this paper is trying to answer is:

- How much can the security techniques GeoIP and fail2ban lower the probability of IoT devices getting hacked?

[2]Source: Kasperysky lab, https://securelist.com/new-trends-in-the-world-of-iot-threats/87991/ accessed 2019-02-22

[3]Source: OWASP, https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project accessed 2019-02-19

[4]Source: https://securelist.com/new-trends-in-the-world-of-iot-threats/87991/ accessed 2019-02-22

Our hypothesis is that it is possible to lower the probability of IoT devices hacked. There probably exists methods that are feasible to implement for IoT manufactures that can reduce the number of successful attacks against IoT devices significantly.

1.2 Related work

There are a few projects that have looked at implementing security techniques for IoT devices, such as: IoT firewall [3], in-hub security manager [4], encryption for IoT devices [5], SVELTE real time IDS [6], and physical layer security [7].

Most of the mentioned work focus on security features that are relatively rarely exploited in the real world. This paper focus on protection against the most common real world attacks. As shown in the introduction most attacks are brute force attacks against SSH and Telnet, which is something fail2ban protect against. The introduction also highlighted that the majority of attacks are coming from only 5 countries. By using a GeoIP security feature a IoT device can protect themselves from all attacks outside a specific country, and therefore shield itself against a majority of the attacks on the internet.

There are also papers that handle the subject of how honeypots can be used as a security feature. How it can trap attacks and prevent them from spreading further. So they can be used for more than to collect data [8].

There exists white papers who either have used or based some of there work on the fail2ban software [9] [10]. There also exists a few projects who have used the Modern Honey Network (MHN) to manage a fleet of honeypots and collect data about attackers [11] [12].

1.3 Scope

This paper focus on implementing two security mechanisms and evaluate their performance. The paper does not try to understand why IoT manufactures does not implement these security features today. Nor does the paper try to understand why IoT manufactures does not change default passwords and does not disable networking services such as Telnet and SSH. The launched honeypots will only be available over IPv4 and not IPv6.

Chapter 2

Background

2.1 Internet of Things

The internet of things (IoT) is a network of many interconnected devices, ranging from single board computers to computers in vehicles, that can share data and information in order to complete various tasks. Its applications are many and the end goal of IoT-devices is to automate and simplify daily tasks. IoT-devices mainly consists of three layers: *Perception layer, network layer* and *application layer.* The perception layer is the layer equipped with sensors and actuators, its task is to detect, gather and process information from its environment. The network layer is tasked with sharing the information acquired by the perception layer across the network it is a part of. This layer can consist of technologies such as Bluetooth, 4G, Wifi etc. depending on the device at hand. The application layer is firstly the one tasked with all authentication, integrity and confidentiality of the data and secondly the layer where the purpose of the IoT-device is specified. This is where the software controlling the device is. It is also this layer where most of the security attacks target[13]. Most IoT-devices use Linux based operating systems and out of them Ubuntu/Ubuntu Core is the most widely used[1].

Since IoT-devices come in different forms, this report is only going to be defining IoT-devices as low-energy devices connected to the internet. For example laptops are not defined as IoT-devices in this report but the computers in a car

[1] Source: https://iot.eclipse.org/resources/iot-developer-survey/iot-developer-survey-2019.pdf accessed 2019-02-22

are since the individual computers are low-energy. These computer are low-energy since they have limited physical space and therefore limited capacity when it comes to CPU and GPU.

2.2 Common attacks

The most common attack for any IoT-device is Distributed-Denial-of-Service(DDoS) attacks. In 2017 a threat landscape reported that around 15% of all attacks were DDoS attacks[2]. DDoS attacks mean that you take down an online service by flooding it with too much traffic beyond its capacity[3]

Another common attack is code execution. The same threat landscape as mentioned above lists it as the second most common attack with around 12% of the attacks[4]. Code execution attacks is where you find a vulnerability so that you can send code that will execute on the attacked machine, meaning the attack can be whatever your code is aimed to do [14]

Brute force is the technique of generating and trying many different combinations of letters, numbers and symbols as passwords to bypass a security wall. For example a lot of IoT-devices are hacked by brute force attacks where many different passwords are generated and tested as the Telnet password and after a while the brute force algorithm generates the right password [5].

Mirai was one of the largest IoT bot nets, internet-connected devices running one or more bots, and it spread to around 300,000 machines in 2016. A study of Mirai showed that a key part for Mirai to spread to so many machines was by trying to connect to new machines with known default credentials [15].

[2]Source: https://www.bitdefender.com/files/News/CaseStudies/study/229/Bitdefender-Whitepaper-The-IoT-Threat-Landscape-and-Top-Smart-Home-Vulnerabilities-in-2018.pdf accessed 2019-05-22

[3]Source: https://us.norton.com/internetsecurity-emerging-threats-what-is-a-ddos-attack-30sectech-by-norton.html accessed 2019-05-22

[4]Source: https://www.bitdefender.com/files/News/CaseStudies/study/229/Bitdefender-Whitepaper-The-IoT-Threat-Landscape-and-Top-Smart-Home-Vulnerabilities-in-2018.pdf accessed 2019-05-22

[5]Source: https://securelist.com/new-trends-in-the-world-of-iot-threats/87991/ accessed 2019-02-22

2.3 GeoIP security feature

Most IoT devices work inside a context where the end users are connecting to device from a nearby region. Think about smart lights, smart TV's, smart toasters. These are all IoT devices that are used in the home of the user. Most users would never try to control their smart TV or toaster from another country. Therefore it makes sense to add a geographical access restriction to the IoT devices.

One way to add a geographical access restriction is by rejecting all IP connection from countries outside a specific country. For specific attacks against you this security feature is not helpful since the attacker can easily use a Virtual Private Network to connect to a server in the country you're in and attack from there, but for automated attacks this can be a great feature. Let's say someone in China finds an exploit that unlocks all doors for locks from a specific country and tries to attack every one of those locks the attack won't even connect to your lock with an GeoIP based security feature since it is an automated attack from China.

2.4 Fail2ban security feature

One of the most common attacks against IoT devices are brute force attacks, it would be desirable to protect against this specific type of attack. Brute force attacks works by repeatably guessing the credentials to the server.

Fail2ban is a software that monitors the log in attempts to the server[6]. With fail2ban one can chose to IP ban a machine that has tried too many times to connect to the server and failed to provide valid credentials.

For instance, one could configure fail2ban to ban every IP that tries to connect to the server three times during the same day.

2.5 Honeypot

A honeypot is a resource set up to monitor unauthorized activity on it. Its value lies in unauthorized uses of it being made since it can collect information about

[6]Source: https://www.fail2ban.org/ accessed 2019-05-06

the security holes in its own system or information about the intruder. Honeypots come in different forms, it can be an application simulating a common service and it can be a network of computers. Honeypots are not intended to produce anything, if it is being used that means that there is an intruder in the system and the set protocols will be followed in that case. For example if the honeypot is a network of computers, the network should not have a production value it is just physically there so that if there is activity on it you can derive that your network has been compromised.[16]

The advantages of honeypots are many, one example is that the data sets they collect are small since they are only active when intruders use them. There are also no false positives nor false negatives, since all activity is unauthorized all activity are negatives. Furthermore since honeypots can come in many different forms they are highly flexible. Also it doesn't matter if the activity is encrypted, since all activity is flagged as unauthorized encrypted activity will be detected.[16]

2.6 Internet Protocol Version 4 (IPv4)

IPv4 is the underlying technique that makes internet connections possible. Every time one connects to the internet one is assigned a unique IP-address and when you want to send information through the internet the information is passed from your IP-address to the address you're trying to connect to. IPv4-addresses are 32 bit addresses meaning there are 2^{32} IPv4-addresses which is about 4.29 billion[7]. Connecting to all IPv4-addresses will take you up to 10 hours for a single ping to each one[8].

2.7 AWS

Amazon Web Services (AWS) is a cloud computing service that among other things rent virtual private servers (VPS) to consumers. A user can create their own VPS with AWS that get assigned a unique IPv4 address. All AWS IP-ranges are public information and can be found on their website[9].

[7]https://www.techopedia.com/definition/5367/internet-protocol-version-4-ipv4

[8]https://www.securityartwork.es/2013/01/21/how-much-does-it-take-to-ping-the-whole-internet-12/

[9]https://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html

Chapter 3

Methods

To answer our research questions, "How much can the security techniques geo-based firewalls and rate limited connections lower the probability of IoT devices getting hacked?", we need to know how many IoT devices are getting hacked without the security techniques and then how many IoT devices would get hacked with the security features. It is impossible to answer exactly how many devices that are getting hacked, since no one can monitor all devices in the world. Therefore the best one can do is to estimate that number. One way to estimate the number of attacks against a machine is by setting up honeypots.

There are a couple of types of hacking attacks that are interesting and practical to measure. The first type is brute force attacks against SSH. These are important to monitor since a big Another important metric to measure is the total number of connections to a machine.

3.1 Launching honeypots

To get an estimate of how many IoT devices that are getting hacked 8 honeypots were launched on AWS. The number of each type of honeypot that were launched is shown in table 3.1. To launch and manage the honeypots Modern Honey Network was used.[1] The honeypots were active for 5 days before they were taken down. All honeypots were launched in AWS' data center "US East (N. Virginia)" and each got one public IPv4 addresses from that region. The machines were not assigned a IPv6 address. The firewall was configured to

[1] Source: https://github.com/threatstream/mhn, accessed 2019-04-10

Honeypot type	#number
Conpot	2
Cowrie	2
p0f	2
Snort	1
Suricata	1

Table 3.1: Honeypots launched

allow connections from everywhere and to all ports. The 8 honeypots were of 5 different types, logging different type of attacks. Some types had two honeypots up to corroborate that the data that was collected was reasonable. The different types were Cowrie, Conpot, p0f, Snort and Suricata. All honeypots where launched on VPS's with a fresh Ubuntu 16.04 LTS.

AWS was used to launch the honeypots due to it's ease of use.

3.1.1 Cowrie

Cowrie is a honeypot that logs all SSH and Telnet connection attempts to the server.[2] For example if someone try to SSH into the server with the user name root and the password admin this would be logged by cowrie. If someone try to exploit a known vulnerability in the server this would not be logged by cowrie.

By launching Cowrie honeypots it is possible to see how many SSH and Telnet connection attempts a server gets. By first launching the Cowrie honeypot without the security features and then with the security features it is possible to see how much the security features reduces the number of attacks.

3.1.2 Conpot

Conpot is a honeypot that logs all attacks directed at Industrial Control Systems (ICS). ICS's is an umbrella term used to encompass different control and monitor systems used in industrial complexes. With the help of a range of frequently used industrial control protocols conpot is capable of emulating big and intricate infrastructure to convince a hacker that they just came across a

[2]Source: https://www.cowrie.org/posts/2015-07-05-cowrie/, accessed 2019-04-10

vast industrial complex.[3]

The relevancy of using this honeypot is that ICS's is an area where IoT-devices are getting more and more integrated. In newer ICS's there is often an IoT platform that connects the different parts of the system and gives users a way of remotely monitor and control the system. Therefore this study will investigate how much the aforementioned security techniques can lower the amount of attacks where ICS's are connected to the internet [17].

3.1.3 p0f

P0f is a honeypot that logs all type of connections to the server.[4] P0f will therefore log all things that all the other honeypots used in this paper also logs. P0f does not log detailed information such as the types of user name and passwords attackers uses to try to log into the system. P0f does not log the type of attacks either.

The benefit of using p0f is that it gives a number of connection attempts to the machine. A machine cannot get hacked without anyone trying to connect to the machine. p0f logs all of these connections attempts. Furthermore, most connections to the honeypots should be from malicious actors and therefore be actual hacking attempts. There are few non-malicious reasons to connect to a random IPv4 address. The most common ones are probably search engines and researchers.

3.1.4 Snort and Suricata

Snort and Suricata are two very similar honeypots. Both honeypots logs events based on specific rules.[5][6] The rules used by this paper are the default rules. These rules check for attackers trying to exploit a wide range of different security holes, such as web specific exploits, malware attacks, common trojans, DNS attacks and SQL attacks among others.

With the data from the Snort and Suricata honeypots it is possible to see how many hacking attempts a machine gets. One problem with Snort and Suricata

[3]Source: conpot.org, accessed 2019-04-10
[4]Source: http://lcamtuf.coredump.cx/p0f3/ , accessed 2019-04-10
[5]Source: https://www.snort.org, accessed 2019-04-10
[6]Source: https://suricata-ids.org, accessed 2019-04-10

is that they only logs down attacks that match a rule. A attacker using a new type of exploit that is not listed in one of the rules, would not get logged. Snort and Suricata might therefore report less hacking attempts than they actually received.

3.2 GeoIP security feature

Iptables is a utility program to set up, maintain and inspect IP filtering rules in the Linux Kernel[7]. Using iptables 'GeoIP' addon, rules to block ip-adresses based on geographical location could be implemented. The GeoIP block used was a rule of blocking all incoming connections from outside the United States since that is where this paper is being conducted. The Maxmind GeoLite2 Country database was used to translate IP addresses to country codes.[8]

3.3 Fail2ban security feature

Fail2ban is a program that can be used to block machines that tries to SSH into a machine multiple times with invalid credentials [9]. Cowrie is the only honeypot launched that monitors SSH log in attempts. Therefore fail2ban is only applicable for Cowrie since the other honeypots does not only monitors SSH log in attempts. Fail2ban version 0.9.3 was used. The rules implemented for the fail2ban used in this paper were instructed to indefinitely ban all IP-addresses that unsuccessfully tried to connect via SSH three times within a 24 hours window.

3.4 Launching honeypots with security features

To estimate how well the two security features would protect against hacking attacks 9 new honeypots where launched on AWS. Each honeypot listed

[7]Source: https://linux.die.net/man/8/iptables, accessed 2019-05-22

[8]Download link to the geolite2 database: https://dev.maxmind.com/geoip/geoip2/geolite2/, accessed 2019-05-01

[9]Source: https://www.fail2ban.org, accessed 2019-05-22

Honeypot type	#number	Security feature
Conpot	2	GeoIP
Cowrie	1	GeoIP
Cowrie	1	Fail2ban
p0f	2	GeoIP
Snort	1	GeoIP
Suricata	1	GeoIP

Table 3.2: Honeypots launched with security features

in table 3.1 was launched again but with the GeoIP security feature enabled, only allowing connections from US IP addresses. One Cowrie instance was launched with the fail2ban feature enabled, protecting against brute force attacks. The exact configurations launched are shown in table 3.2.

Chapter 4

Results

4.1 Honeypots without any security features

The honeypots without any security features were up and collecting data for nearly 5 days, more specifically the machines were live between 2019-04-08 6pm and 2019-04-13 5pm UTC. The results from these honeypots are presented in table 4.1. In total all our honeypots received a cumulative of 100,274 attacks.

Honeypot	Attacks	Attacks / hour
p0f-1	19,134	160.8
cowrie-1	27,060	227.4
snort-1	3,905	32.8
suricata-1	4,235	35.6
conpot-1	3,073	25.8
cowrie-2	20,156	169.4
p0f-2	18,986	159.5
conpot-2	3,509	29.5

Table 4.1: Attacks against honeypots without any security features.

Out of the three honeypots that had two sensors up and running (Conpot, p0f and Cowrie) Conpot and p0f had a very similar amount of attacks for the two sensors. Cowrie on the other hand had one sensor with only approximately 3/4 of the attacks of the other sensor. One of the Cowrie sensors also had a very similar amount of attacks to p0f even thought it detects a specific attack, ssh-

attacks, and p0f detects all connections as attacks. The other Cowrie sensor on the other hand had a significantly higher amount of attacks than any of the other honeypots.

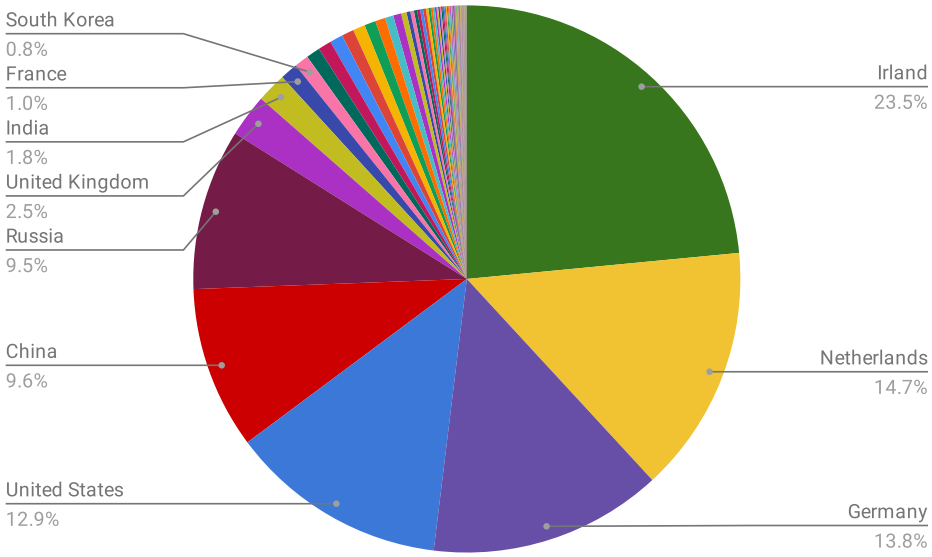

Figure 4.1: Origin of attacks, all honeypots

As seen in figure 4.1 most attacks originate from Ireland, followed by the Netherlands and Germany. In total these countries contributed to more than 50 % of the attacks. However, most of the attacks from these countries where all targeted against the cowrie honeypots. Figure 4.2 shows the breakdown of attacking countries to all honeypots but cowrie. Most attacks are now coming from the US (24 %) followed by Russia (18 %) and China (18 %). There was in total 53,048 attacks against all our honeypots excluding cowrie.

The breakdown of attacks against the industrial system honeypot conpot is vastly different too, as seen in figure . Most of the attack against conpot originate from China (55 %) followed by Hong Kong (7 %). There was 6,719 attacks in total against the conpot honeypots.

The average number of attacks per IP is shown in table **??**. Cowrie, which logs SSH attempts, have a significant higher number of attacks against it from the same IP's. That is, once an attacker from a specific IP finds a cowrie honeypot, they are more likely to keep attacking from the same IP, compared to attackers who attack the other honeypots. Note that suricata, p0f and snort all have a similar number of average attacks per IP, which is expected since

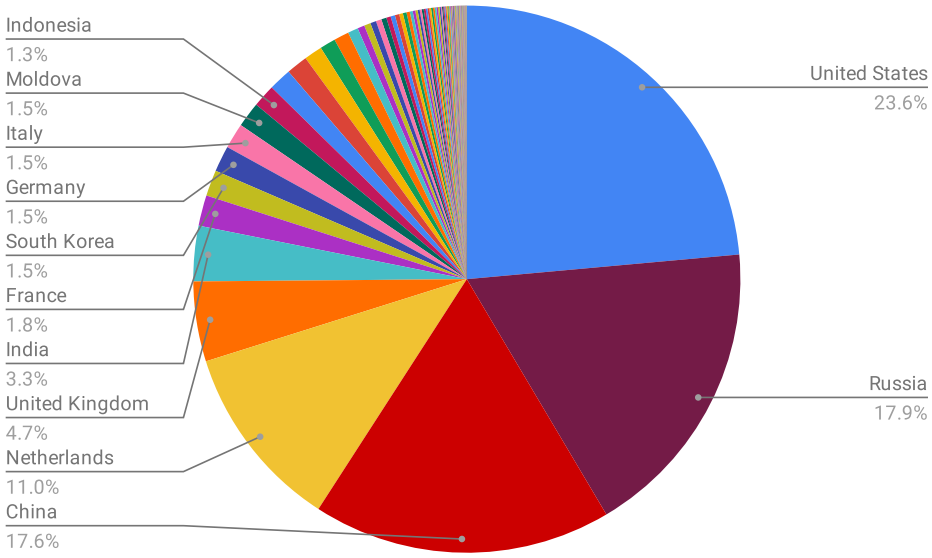

Figure 4.2: Origin of attacks, all honeypots but cowrie

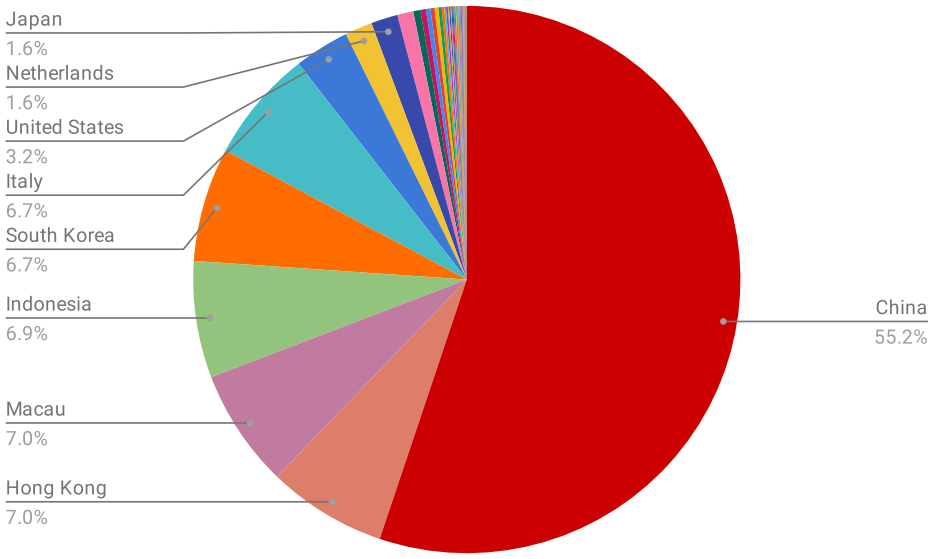

Figure 4.3: Origin of attacks targeting the conpot honeypots

these honeypots works in a similar fashion. Conpot, which simulates common industrial control systems, got on average 17 attacks from each IP.

4.2 Honeypots with security features

The honeypots with security features were up and collecting data for also nearly 5 days, more specifically the machines were live between 2019-04-24 6pm and 2019-04-29 5pm UTC. The results from these honeypots are presented in table 4.2. In total all the honeypots with security features received a cumulative of 6,184 attacks. This is a 94 % reduction in attacks.

Honeypot	Security Feature	Attacks	Attacks / hour
p0f-1	geo	2,301	19.3
cowrie-1	geo	36	0.3
snort-1	geo	301	2.5
suricata-1	geo	289	2.4
conpot-1	geo	269	2.3
cowrie-2	fail2ban	189	1.6
p0f-2	geo	2,647	22.3
conpot-2	geo	152	1.3

Table 4.2: Attacks against honeypots with security features.

4.3 Comparison

A direct comparison of the results with and without security is presented in table 4.3, table 4.4 and figure 4.4.

For all of the honeypots the amount of attacks decreased by at least 87%, some decreased by 99%. Note that the Cowrie honeypot with the Geo IP security feature has a very few amount attacks compared to any other honeypot.

One can also see that the honeypot with the most average number of attacks per IP address also has the biggest reduction in attacks per IP address.

Honeypot	Avg w/o Security Feature	Avg with Security Feature	Attack Reduction
p0f (geo)	19,060	2,474	87.0%
cowrie (geo)	23,608	36	99.8%
cowrie (fail2ban)	23,608	189	99.2%
conpot (geo)	3,291	210	93.6%
snort (geo)	3,905	301	92.2%
suricata (geo)	4,235	289	93.2%

Table 4.3: Comparison for the honeypots before and after security features were implemented

Honeypot	Avg number of attacks per IP (before)	Avg number of attacks per IP (after)
cowrie (geo)	101.1	1.7
cowrie (fail2ban)	101.1	4.6
conpot	17.4	12.8
suricata	7.3	1.9
p0f	6.5	6.5
snort	6.5	1.8

Table 4.4: Average number of attacks per IP against each honeypot type

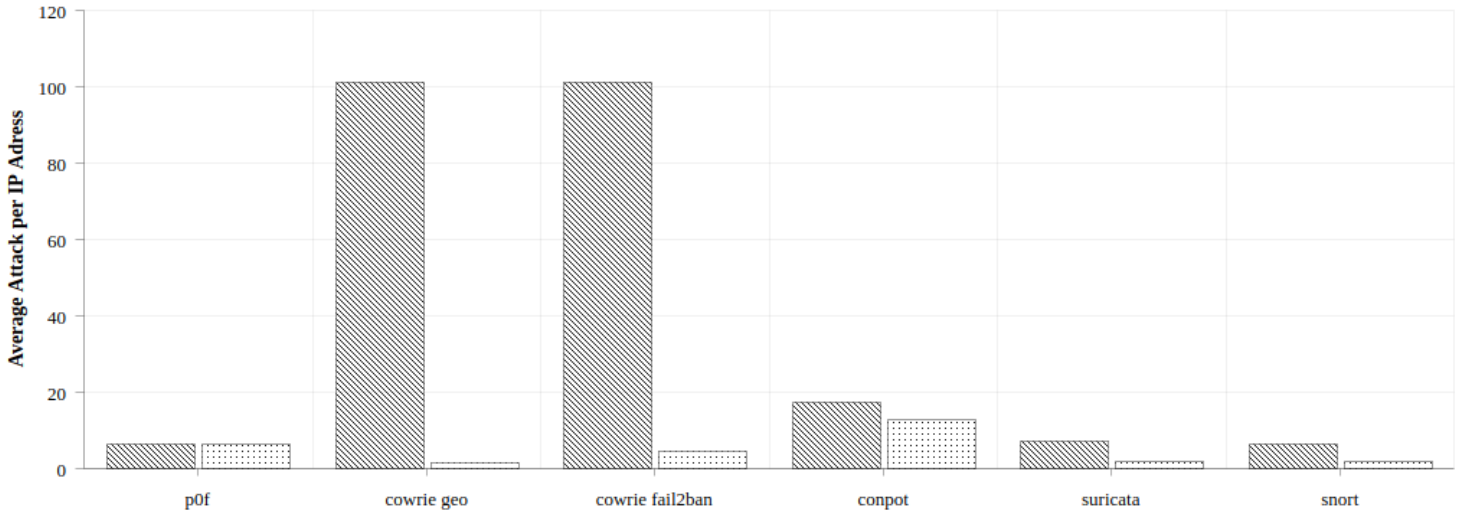

Figure 4.4: Average Attack per IP address before and after security implementations

Chapter 5

Discussion

5.1 GeoIP effectiveness

The results shows that the security technique GeoIP can reduce the number of attacks by between 87.0% - 99.8%. The reduction in attacks against p0f was 87%. P0f is a honeypot that logs all incoming connections to the machine. Not all of these connections will be malicious hacking attempts. Some of the connections could come from search engines that tries to index the internet. Note that web search engines such as Google and Bing would not crawl our honeypot since our server does not host any web server. The IP's to our honeypots are not listed anywhere on the web either. Researchers are another group of people who might connect to random IPv4 address on the internet. Besides these two groups, search engines and researchers, there are probably not many non-malicious reasons to connect to random IPv4 addresses on the internet. The number of connections from search engines and researchers are probably very low too. Therefore one can assume that almost all of the connection attempts to the honeypots are made with a malicious intent.

Snort and Suricata are honeypots that only monitors actual potential attacks. A flaw with these honeypots are that they determine whether a connection is an attack or not by matching the connection against a list of rules. If the connection does not match any of the rules the connection is not marked as malicious. The list of rules cannot cover all potential attacks. For example, new attacks that are developed after the list of rules is installed, cannot possibly be detected, since the new attack is not a part of the old list of rules. The number of attacks against Snort and Suricata was reduced by 92% and 93%. This is sim-

ilar to the results for p0f, who got a 87% reduction of attacks. Conpot which is a honeypot that only logs specific attacks against industrial systems also got a similar reduction in number of attacks, 94%. The effectiveness of the GeoIP security features seems to be around 90 % for a US based server.

Furthermore, one can assume that the relation between the number of hacking attempts and the first successful break-ins are linear. The assumption is that there on average takes some number of random attacks from bots on the internet before one bot successfully manage to break-in to the machine. Say for example that it on average takes 100,000 random connections before a US based machine becomes compromised. Also assume that the machines gets 1,000 malicious connections per day. It would then take 100 days on average before the machine is compromised. If the machine would have the GeoIP security feature installed that according to the result chapter reduces attacks by 90%, it would now take 1,000 days before the machine becomes compromised by an automated bot.

The GeoIP security feature can therefore prolong the time it takes before a machine becomes compromised by 10 times.

5.1.1 GeoIP relevance for IoT

It is reasonable to think that GeoIP has not been widely adopted because traditional servers usually wants to accept connections from everywhere. A web server wants to accept connections from the entire world. Even if the web page hosted on the server only is intended for US customers you might have US customers who are traveling in Europe and accessing the website. Furthermore the server probably do not want to block potential web crawlers from search engines whose servers are located outside of the US.

However, with IoT devices the intended users are often in the same city as the IoT device. Think about a Smart TV, a smart light bulb, a smart toaster, a baby monitor camera and other similar home IoT devices that are increasing in popularity. These are all devices that the end user want to be able to control while they are being home, maybe on their way to and from work, and at work. But the amount of users who wants to control their smart devices at home when they are traveling on vacation is probably very low. Therefore it makes a lot of sense to apply geographical IP access restrictions to IoT devices. In a lot of cases no one from outside the country or even the city needs to connect to the IoT device.

The same logic applies to IoT devices such as industrial control systems used in the industry. These devices usually just need to be accessed by the workers in that factory. They might have a need to be able to read statuses and control the machines remotely from their homes and on the way to and from work. But the workers do not need to access the machine if they go on a vacation to for example China.

One important limitation to the GeoIP security feature is that it only protect against automated attacks. If a malicious actor wants to hack a specific machine with the GeoIP security feature enabled, they can easily by pass the feature by using a VPN from a country that is white listed by GeoIP.

5.1.2 Average number of attacks per IP

One would assume that the average number of attacks per IP should be the same with the GeoIP security feature. However in table 4.4 we can see that the average was reduced for all honeypots but p0f. The reduction in average attacks per IP could indicate that attackers from the US for some reason try fewer exploits in there attacks. It could also indicate a potential flaw in the method. The number might had been more similar before and after the GeoIP security feature if the honeypots had been live for a longer time period than five days. All honeypots but p0f received a relatively small number of attacks as seen in table 4.3. p0f had 2474 attacks with the security feature and all other honeypots had less than 302 attacks. These relatively small numbers could lead to skewed average numbers.

5.2 Fail2ban effectiveness

The results showed that the average attack per IP-address for the Cowrie honeypot was notably higher than the numbers for other honeypots. This can be explained by the fact that one of the most usual attacks, as mentioned in the background, is SSH-brute force attacks and Cowrie logs SSH-attacks. This is the very reason that this paper looked to implement fail2ban for the Cowrie honeypot and the results showed that it had an effect in decreasing the amounts of attacks. Attacker who try to brute force the SSH credentials are essentially connecting to the machine, trying a user name and password combination and if that combination was wrong they connect again using a different combina-

tion. By the very nature of this type of attacks one would expect an attacker to try a lot of different combinations, therefore it is expected that the average number of attacks per IP is much higher than for the other honeypot types. First of all the amount of attacks went down by 99.2% and second of all the average attacks per IP-address went from 101.1 to 4.6 which is in line with the fail2ban rules which stated that IP-addresses would get banned after 3 unsuccessful tries in a 24-hour window. That the average is higher than the max amount could be explained by the fact that it takes some time between an SSH-connection and the actual ban happens, so some IP-addresses might have had time to do more than 3 attempts before they got banned.

It is hard to say exactly how effective the fail2ban security feature is at protecting a machine against brute force attacks. It is clear from the results that fail2ban reduced the number of SSH-connection attempts by 99%. However, this does not mean that the risk for credentials getting leaked is reduced by 99%. A lot of machines on the internet uses default credentials. Combinations such as "root/root" are probably very common. Common combinations of credentials are probably tested in the first attempts by automated bots who try to brute force into other machines. If a machine is using such a weak user name and password combination as "root/root" fail2ban would not help to protect the machine, since a hackers first guess probably would be "root/root".

5.3 Method

In this paper VPSs were used through AWS to deploy the honeypots. The IP-ranges for AWS are public information easily accessible on their website. This could lead to the results being a bit skewed since the honeypots were all deployed on IP-addresses that are well known. But for two reasons the results shouldn't be skewed to an extent to where they are unusable. First of all, they were all assigned IPv4 addresses and all IPv4 addresses can be pinged within 10 hours, so even if the IP address is not well known it will get detected by automated bots. The second reason why this shouldn't skew the results to an unusable extent is that the honeypots with and without the security features were deployed on AWS servers meaning that both results would be skewed and therefore the percentage differences should be accurate.

In this paper it is assumed that the VPSs used and IoT-devices are equivalent for a few reasons. By earlier definition IoT-devices are physical devices connected to the Internet. The security features used did not try to protect against

hardware attacks but was all focused on the attacks that would come through the Internet and therefore VPSs connected to Internet would face the same attacks. Furthermore the operating system used on the VPSs launched was Ubuntu 16.04 and a lot of IoT-devices uses Ubuntu and similar Linux based operating systems. So the protocols and features of the devices and the VPSs would be very similar. Lastly the security features added in this paper protected against attacks that are universal across all operating systems. SSH is used in almost all operating systems so investigating ssh-attacks and how one can limit them will be something a lot of IoT-devices can make use of and the attacks will come from all over the world as shown in the results so using GeoIP based security can protect regardless of operating systems.

5.4 Future work

While conducting this paper and trying to implement the GeoIP based security features a lot of problems within that process stood out as unreasonably complicated. First of all the installation process of the GeoIP module had a lot of errors and needed fixes for many of the steps. Second of all, and maybe the biggest problem, the entire system got a lot slower after implementing the GeoIP module that banned every IP address outside of the United States, commands that should take microseconds took about 10 seconds. It might have something to do with the GeoIP module having a very slow look up when cross referencing an IP-address to something that is banned and the amount of those IP addresses can be up to 2^{32} that is 4.29 billion. But if the data structure holding that information was structured in an efficient way and sorted, it should only take up to 32 computational steps with a $O(log(n))$ look up algorithm and only 1 step with a $O(1)$ algorithm in a structure using a hash function, so there is some room for improvement to develop a more effective GeoIP module.

Another area for future work is to investigate how much the GeoIP and fail2ban security feature can slow down the propagation of a virus, such as a bot net. If a large amount of the computers on the internet would have GeoIP enabled it would take longer time before a virus is able to spread to another country, since the virus would need to find a computer without GeoIP enabled. In a similar manner it would take longer time for a virus to brute force the credentials of servers with fail2ban installed, since they only can try a very limited number of credentials before the IP gets banned.

To research how much these security features would slow down the transmission of a virus the researches could simulate two copy's of a subset of the internet. In copy one some of the computers would have the aforementioned security features enabled. The researchers would deploy a virus in both copy's of the internet and compare how fast the virus propagates in the subset with security features versus the copy without any security features.

Lastly, future researchers can try to statistically verify the results presented in this paper or verify new results following the same methods. Some numbers did stand out as a bit odd and it would be interesting to see how they would be explained in a statistical context. For example, only 3.2% of the attacks against Conpot are from the US in the data collected before the security features were implemented leading one to believe that the decrease would be around 97% of the attacks but the actual result was a 93% decrease, the decrease was less than expected. For the honeypots without cowrie it was the other way around, 23.6% of the attacks where from the US before any security feature was implemented leading one to believe that the decrease would be around 74% but the actual decrease was 92%. It would be interesting if these numbers could be explained using statistical methods.

Chapter 6

Conclusions

The research questions is "How much can the security techniques GeoIP and fail2ban lower the probability of IoT devices getting hacked?". The GeoIP security technique can lower the probability of an IoT devices getting hacked by roughly 90 %. The fail2ban security technique reduces the risk of a machine getting their credentials brute forced by 99 %. These two security techniques heavily reduces the number of attacks and therefore the probability of an IoT device getting compromised.

Since the security features are effective at reducing the probability of getting an IoT device hacked and these features are transparent to the intended users of the IoT devices, these security features should be enabled by default. These features should come pre-installed and enabled in IoT operating systems and IoT frameworks.

Future work in this field is encouraged. More work needs to be put into making a GeoIP security feature that is easier to install and making the program faster. Research about how effective GeoIP and fail2ban would be at slowing down the propagation of viruses on a large scale would also be beneficial to better understand more precisely the potential impact these security features can have on making the internet more secure.

Bibliography

[1] Mozilla. *Internet Health Report 2018*. Tech. rep. URL: https://internethealthreport.org/2018/.

[2] Constantinos Kolias et al. "DDoS in the IoT: Mirai and other botnets". In: *Computer* 50.7 (2017), pp. 80–84.

[3] N. Gupta, V. Naik, and S. Sengupta. "A firewall for Internet of Things". In: *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*. Jan. 2017, pp. 411–412. DOI: 10.1109/COMSNETS.2017.7945418.

[4] Anna Kornfeld Simpson, Franziska Roesner, and Tadayoshi Kohno. "Securing vulnerable home IoT devices with an in-hub security manager". In: *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*. IEEE. 2017, pp. 551–556.

[5] D. Altolini et al. "Low power link layer security for IoT: Implementation and performance analysis". In: *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*. July 2013, pp. 919–925. DOI: 10.1109/IWCMC.2013.6583680.

[6] Shahid Raza, Linus Wallgren, and Thiemo Voigt. "SVELTE: Real-time intrusion detection in the Internet of Things". In: *Ad hoc networks* 11.8 (2013), pp. 2661–2674.

[7] Yuanyu Zhang et al. "On secure wireless communications for IoT under eavesdropper collusion". In: *IEEE Transactions on Automation Science and Engineering* 13.3 (2016), pp. 1281–1293.

[8] Feng Zhang et al. "Honeypot: a supplemented active defense system for network security". In: *Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies*. IEEE. 2003, pp. 231–235.

[9] James Yu. “An Empirical Study of Denial of Service (DoS) against VoIP”. In: *2016 15th International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security (IUCC-CSS)*. IEEE. 2016, pp. 54–60.

[10] Florin B Manolache, Qingping Hou, and Octavian Rusu. “Analysis and prevention of network password guessing attacks in an enterprise environment”. In: *2014 RoEduNet Conference 13th Edition: Networking in Education and Research Joint Event RENAM 8th Conference*. IEEE. 2014, pp. 1–7.

[11] Hibatul Wafi et al. “Implementation of a modern security systems honeypot Honey Network on wireless networks”. In: *2017 International Young Engineers Forum (YEF-ECE)*. IEEE. 2017, pp. 91–96.

[12] Chris Moore and Ameer Al-Nemrat. “An analysis of honeypot programs and the attack data collected”. In: *International Conference on Global Security, Safety, and Sustainability*. Springer. 2015, pp. 228–238.

[13] Rwan Mahmoud et al. “Internet of things (IoT) security: Current status, challenges and prospective measures”. In: *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. IEEE. 2015, pp. 336–341.

[14] Arvind Seshadri et al. “Pioneer: verifying code integrity and enforcing untampered code execution on legacy systems”. In: *ACM SIGOPS Operating Systems Review*. Vol. 39. 5. ACM. 2005, pp. 1–16.

[15] Manos Antonakakis et al. “Understanding the mirai botnet”. In: *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 2017, pp. 1093–1110.

[16] Lance Spitzner. “Honeypots: Catching the insider threat”. In: *19th Annual Computer Security Applications Conference, 2003. Proceedings.* IEEE. 2003, pp. 170–179.

[17] Ioan Ungurean, Nicoleta-Cristina Gaitan, and Vasile Gheorghita Gaitan. “An IoT architecture for things from industrial environment”. In: *2014 10th International Conference on Communications (COMM)*. IEEE. 2014, pp. 1–4.

TRITA-EECS-EX-2019:395