How The Internet of Vulnerable Things Played a Role in DDoS Attacks in Late 2016

Computer Security Case Study

Jacob House
201614260

Submitted On: *9th August 2019*

Submitted To:
Dr. Jonathan Anderson
Faculty of Engineering and Applied Science
Department of Electrical and Computer Engineering

Course Name: *Engineering 7864 – Computer Security*

Contents

1 Introduction

Nearly two decades into the twenty-first century, the phrase "information age" is evolving into something that encompasses more than just information itself. Modern technology enables the world as we know it today, where information is available instantly, at the touch of a button, from any device, in nearly any location. Such convenience has resulted in the growth of connected devices that were once only toys of hobbyists: the Internet of Things, commonly abbreviated to IoT.

These small, internetworked, "smart" devices are appearing in all facets of modern life; they may be seen in fields varying from agriculture to electrical grid monitoring to home automation — even nano-scale and implantable medical devices — and the list only continues to grow. In fact, the International Data Corporation (IDC) reports as recently as January of this year that spending on IoT is forecasted to reach nearly $750 billion by the end of 2019. This is up 15.4% over the $646 billion spent last year [1]. Further, IDC predicts that IoT spending will sustain its double-digit annual growth rate through the year 2022, in which it anticipates an annual expenditure of $1.2 trillion [2]. In other words, the 6–9 billion IoT devices that were on the Internet in 2016 is expected to grow to about 30 billion by 2020 [3].

2 Background: IoT Growth and Password Misuse

As the IoT world continues to grow, so too does its attractiveness to cyber criminals as a tool to launch attacks. In particular, the distributed nature of a global network of IoT devices under an attacker's control (*i.e.,* a botnet) allows such an attacker to launch distributed denial of service (DDoS) attacks with unheard of amounts of traffic hitting the target from all over the globe. In fact, the "primary purpose of IoT malware is DDoS attacks" [4].

2.1 The Role of Default Passwords

In order for such an attack to take place, attackers may leverage the fact that many IoT devices are deployed with default or near-default settings. More importantly, this includes default administrative or management credentials. According to [5], "many services are designed with default passwords to bypass authentication to provide immediate, temporary access for quick, convenient initial set up of infrastructure". The paper emphasizes, however, that such credentials should only be used for initial set up; they should be changed as soon as possible.

Unfortunately, several studies have pointed out that many Internet-facing services and devices utilize default administrative credentials [5, 6].

2.2 A Trend of Password Misuse

Inglesant and Sasse of the University College London point out that "users are in general concerned to maintain security [through the use of appropriate passwords]", however "password policies that do not meet users' work practices caused high levels of dissatisfaction, and led to insecure practices and low security motivation" [7]. Consequently, there have been a number of cyber attacks resulting from use of insecure passwords. In particular Knieriem et al. note a 2012 attack on the Utah Department of Health (UDOH) which resulted in the loss of 780,000 Medicaid patient health records and over 255,000 social security numbers as a result of default passwords giving attackers complete access to the servers in question [5, 8].

Approximately a month prior to the acknowledgment of the UDOH breach, the US Department of Energy published the results of a security audit on IT systems at the Bonneville Power Administration which provides about 30% of wholesale power to regional utilities in the US Pacific Northwest [9]. The report discusses testing that "identified eleven servers that were configured with weak passwords, an issue that could have allowed a knowledgeable attacker to obtain complete access to the system." Further, "four servers were configured to allow any remote user to access and modify shared files" and "at least one administrative account [was found] with a default password" [9, 10].

Cases such as these allow researchers to paint a picture of flawed and nonexistent password use being a widespread issue affecting enterprise, government, and consumers.

2.3 Default Passwords Pave the way for Mirai

With poor password practices being as provably widespread as our findings have observed, it is not surprising that Fraunholz et al. found that automated scanning for exposed services targeted common username-password combinations. Through the use of medium-interaction honeypots[1] (MIHP)

[1] Honeypots are monitored computer systems designed to mimic cybercriminals' targets. Through simulating the environment of a real victim computer, security and digital forensic analysts are able to observe malicious intrusion in real-time and collect detailed logs of the attacker's interaction

exposed using the SSH and Telnet protocols for 106 days, Fraunholz et al. were able to observe and classify 156,772 attacks. Of these, over 46,000 login attempts used the username `root`. Other common attempts were made to accounts named `admin`, `guest`, and `administrator` with quantities of 13,285, 1,951, and 1,659, respectively. Common password attempts included `root`, `1234`, `xc3511`, among others [6]. The presence of less generic username-password pairs such as `root/xc3511` which occurred 4,019 times (as opposed to `root/root`, for example) indicate that the observed scanning is targeting a particular service or device model. This username-password combination is indicative of scanning targeting DVR and IP camera models made by the Chinese manufacturer XiongMai Technologies [11]. Fraunholz et al. claims that through such classification techniques they were able to deduce that about a third of all login attempts were targeting specific embedded devices such as routers, DVRs, or IP cameras.

This type of highly specialized scanning became known in mid- to late-2016 as the modus operandi of the Mirai botnet. Mirai infects embedded and IoT devices — specifically IP cameras and DVRs — and is used to launch DDoS attacks as described at the beginning of this section. In September and October 2016, Mirai launched DDoS attacks towards three major targets:

(i) hosting provider OVH (901 Gbps first attack, 1.1 Tbps second attack [12]),

(ii) security blog *Krebs on Security* (generating 623 Gbps of traffic [4, 12, 13]),

(iii) DNS provider Dyn

These attacks crippled popular Internet sites including Netflix, PayPal, Twitter, Shopify, Airbnb, Github, Reddit, Kayak, *The New York Times*, *The Wall Street Journal*, and Box [3, 5, 12–14]. According to [12], "the assault was so effective — and sustained — that Krebs' longtime DDoS mitigation service, Akamai, one of the largest bandwidth providers on the internet, announced it was dropping Krebs' site because it couldn't bear the cost of defending against such a massive barrage". The attack on Dyn was so large that the provider later announced that it might never be able to calculate the full weight of the assault it faced: "There have been some reports of a magnitude in the 1.2 Tbps range; at this time we are unable to verify that claim" [12, 15–17].

Such astronomical measures of traffic were possible because of the number of devices from which traffic originated [13].

with the supposed victim computer. This may be useful to make predictions and to observe trends relating to attack origin, type, intention, etc.

3 DNS Water Torture as a Means of Bandwidth Depletion

Mirai is capable of launching a number of different types of denial of service attacks [13]. This study will focus on DNS floods.

3.1 DNS Query Analysis

When computers wish to communicate with other hosts, they often do so using the remote host's domain name, such as `www.google.com`, which must be resolved to an IP address. This is done using the domain name system (DNS).

Often computers are configured to query recursive DNS resolvers owned by an ISP or other Internet service. Recursive DNS resolvers are resolvers that perform iterative lookups on behalf of clients. For example, if a computer wishes to know the IP address of `www.cl.cam.ac.uk`, the recursive resolver would ask the authoritative nameserver for `.` which nameserver is authoritative for `.uk.`. This nameserver would then be queried to determine what nameserver is authoritative for `.ac.uk.`, and so on, until the recursive resolver has the full answer containing the IP address of `www.cl.cam.ac.uk.`. This result is then sent to the client.

This mechanism may be abused by repeatedly asking numerous recursive resolvers for bogus or non-existent subdomains of a particular domain. For each "fresh" query (*i.e.,* each new non-existent subdomain), the recursive resolver will need to converse with the authoritative nameserver for the victim domain. When large numbers of hosts make such requests simultaneously, the authoritative nameserver for the victim's domain is unable to keep up with all of the requests and can no longer service legitimate requests for real subdomains.

3.2 The Affect of TTL on DDoS Victims

DNS responses are accompanied by a time to live (TTL) which states how long the response may be cached and considered valid for. When the TTL expires, nameservers must query the authoritative nameservers again. As a result, short TTL values allow DNS changes to be very quickly propagated, while long TTLs result in less load on authoritative nameservers since responses are cached for longer amounts of time.

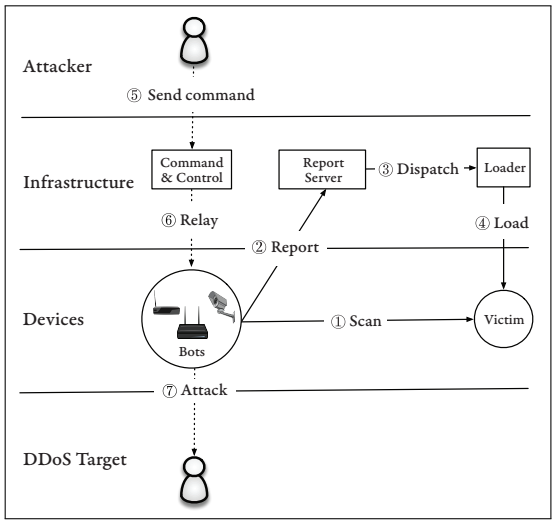


Figure 1: Anatomy of the Mirai Botnet *(figure from [13])*

In the case of DNS flooding, a short TTL may cause a denial of service attack to be much more effective as caches containing legitimate subdomains expire very quickly. In the event of Mirai attacks, websites with 24 hour TTLs were likely almost unaffected even as DNS providers were crippled by the floods of traffic they received. Twitter, on the other hand, had a TTL of 205 seconds and was therefore taken offline within four minutes of the DDoS attack beginning [18].

4 Spread of Infection

The malware's toolkit consists of four main entities designed to optimize the spread of infection: (i) infected devices, (ii) a central command and control (C2) server, (iii) a report server, and (iv) a loader (Figure 1).

The spread of the malware begins with already-infected devices that perform scanning of other Internet hosts on TCP ports 23 and 2323, used by the antiquated Telnet protocol (#1 in Figure 1). The devices attempt to log on using ten random username-password combinations from a list of 62 default credentials, some of which were specific to the devices being targeted — such as the `root/xc3511` pair [13]. Others were more generic, such as `root/root` and `admin`/[blank]. Upon successful authentication, bots report the vulnerable device to the reporting server which asynchronously triggers a loader (#3 in Figure 1) to infect the new victim device (#4 in Figure 1) [13].

4.1 Available Reconnaissance Tools

To observe the growth patterns of malware like Mirai, Antonakakis et al., Hallman et al., and Bailey et al. observed interactions between Internet bots and monitored computer systems [6, 13, 19, 20]. These research efforts varied in execution with Antonakakis et al. using data from Merit's network telescope [13, 19], Hallman et al. using medium-interaction honeypots [6], and Bailey et al. using an Internet Motion Sensor (IMS) similar to Merit's network telescope [20].

In [19], Merit defines a network telescope — also known as a Darknet — to be "an unused, but routed, IP address space that is utilized for the collection and analysis of unsolicited internet traffic." As these IP spaces are unused, no legitimate hosts exist in the address block. Consequently, traffic destined for such networks "must be the result of misconfiguration, backscatter from spoofed source addresses, or scanning from worms and other probing" [20].

Bailey et al. point out that there are two primary challenges faced by network telescopes and IMSs (hereinafter *monitors* refers to both Merit's network telescope and Bailey et al.'s IMS). To collect sufficient data, monitors must have a large enough surface area to be visible to Internet threats. In addition to size requirements, Bailey et al. has found that "address blocks in different networks see different thread traffic." This means that an effective monitor must cover not only a large amount of IP space, but must also have sufficient topological coverage of the Internet. To accomplish this, Bailey et al.'s IMS used 28 distinct monitored subnets in 18 physical locations. These subnets ranged in size from /25 to /8 (*i.e.,* 126 hosts to 16,777,214 hosts) and spanned "major service providers, large enterprises, academic networks, and broadband providers" [20]. Merit's network telescope contained 4.7 million IP addresses and monitored traffic from mid-July 2016 to late-February 2017 [13].

4.2 Identifying Mirai Traffic

Antonakakis et al.'s network telescope received an average of 1.1 million IP packets from 269,000 hosts per minute throughout its monitoring period [13]. Not all of this traffic originated from Mirai scanning and hence needed to be filtered. The study make the following remark concerning filtering methods:

> To distinguish Mirai traffic from background radiation and other scanning activity, [Antonakakis et al.] uniquely fingerprinted Mirai probes based on an artifact of Mirai's stateless scanning whereby every probe has a TCP sequence number — nor-

> mally a random 32-bit integer — equal to the destination IP address. The likelihood of this occurring incidentally is $1/2^{32}$, and [researchers] would expect to see roughly 86 packets demonstrating this pattern in [their] entire dataset. In stark contrast, [they] observed 116.2 billion Mirai probes from 55.4 million IP addresses. Prior to the emergence of Mirai, [they] observed only three IPs that perform scans with this fingerprint. [13]

The researchers point out that, regarding the 269,000 hosts per minute quoted above, raw counts of source IPs was found to be a poor metric of botnet size due to DHCP churn, in which IP addresses are redistributed by a DHCP server. This results in IPs that were counted as being sources of Mirai traffic being released and then leased to non-malicious hosts. The compromised hosts then obtain new IP addresses and are counted again as infected source IPs, resulting in an inflated quantity of infections being recorded. To minimize the number of false positives, Antonakakis et al. tracked the size of the botnet by considering only the number of hosts probing their network telescope at the start of every hour.

Fraunholz et al. compared attacker IPs with Tor exit nodes, free proxies, and known VPN servers and found no matches. They believe this to be an indicator that large amounts of detected traffic was coming directly from the botnet's automated propagation mechanisms since they "usually do not use such means of obfuscation."

4.3 Watching Mirai Grow

Using Merit's network telescope data, Antonakakis et al. were able to determine that within its first 20 hours on the Internet Mirai infected nearly 65,000 devices (Figure 2a on the following page), doubling in size every 76 minutes from a single IP address belonging to DataWagon, a U.S. bulletproof hosting center provider, before reaching a steady state population of between 200,000 and 300,000 infections with a peak of an estimated 600,000 infected hosts [13]. Note that within the first two days Mirai's Telnet scanning is on-par with that of all existing non-Mirai scanning.

The spike to 600,000 scans on November 26, 2016 in Figure 2b on the next page is a result of additional protocols being scanned by bots as various strains of the malware emerged (see Section 5). In particular, Antonakakis et al. notes that this spike corresponds to "Mirai compromis[ing] CWMP[2] devices through an RCE exploit in a SOAP configuration endpoint". This strain of the malware targeted routers and led to an outage at Deutsche Telekom in late November. This resulted in a surge of CWMP (TCP port 7547) scanning which then died down as Deutsche Telekom patched affected routers soon after the attacks [13].

[2] CPE WAN Management Protocol (CWMP) is a HTTP-based protocol that enables auto-configuration and remote management of home routers, modems, etc. [13].

4.4 Juxtaposition Between Attacker and Victim Geography

Using Maxmind[3], Antonakakis et al. were able to map where scans and attacks were coming from *geographically*, and where they were destined. Using this information, the researchers were able to compare countries that harboured the most infections on September 21, 2016 (when the attack on *Krebs on Security* occurred) with countries that hosted the most Telnet devices prior to Mirai's onset. They found that a disproportionate number of infections occurred in South America and Southeast Asia, accounting for 50% of all infections (Table 1).

Country	Mirai Infections	Mirai Prevalence	Telnet Prevalence
Brazil	49,340	15.0%	7.9%
Colombia	45,796	14.0%	1.7%
Vietnam	40,927	12.5%	1.8%
China	21,364	6.5%	22.5%
S. Korea	19,817	6.0%	7.9%
Russia	15,405	4.7%	2.7%
Turkey	13,780	4.2%	1.1%
India	13,357	4.1%	2.9%
Taiwan	11,432	3.5%	2.4%
Argentina	7,164	2.2%	0.2%

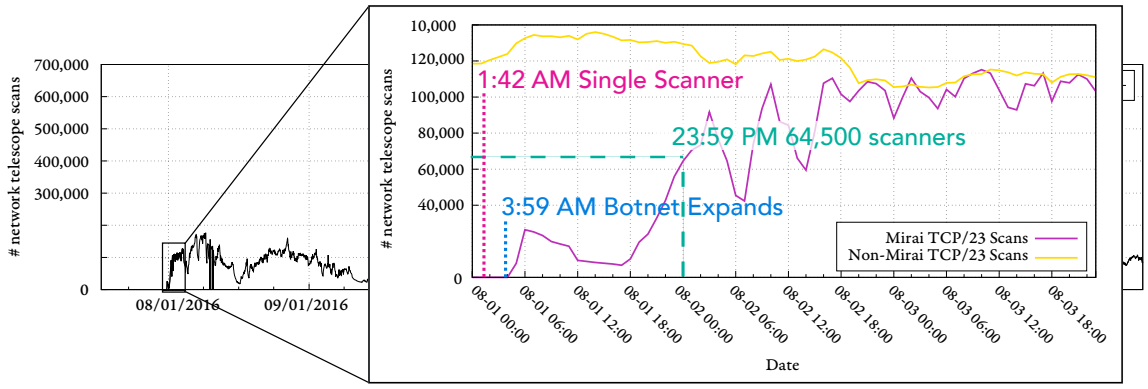
Table 1: Geographic Distribution of Mirai Infections *(table from [13])*

These findings contrast with the fact that Mirai victims were distributed across 85 countries with 50.3% being in the United States, 6.6% being in France, and 6.1% being in the United Kingdom [13].

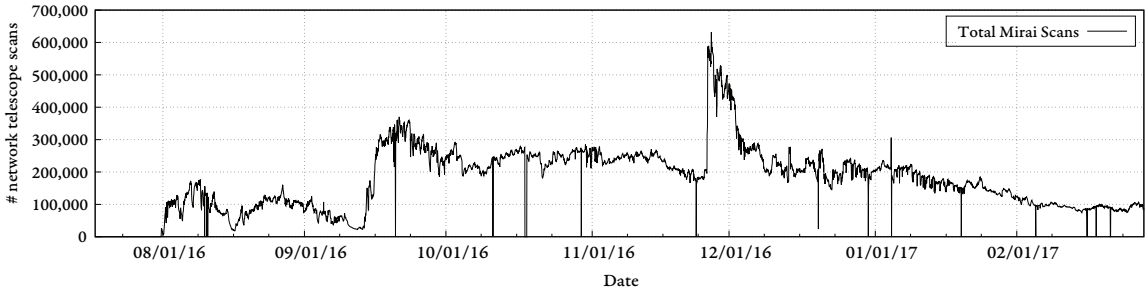
4.5 Telnet Honeypot Findings

Both Fraunholz et al. and Antonakakis et al. used honeypots to record the botnet's capabilities as the malware evolved (see Section 5). The former observed that upon infection, `sh` and `/bin/busybox` were the top two commands issued by the malware on newly-infected devices. The latter pre-

[3] `https://www.maxmind.com/en/geoip2-city`



(a) Rapid Emergence



(b) Steady State Population

Figure 2: Mirai Botnet Population *(figures from [3])*

sented the BusyBox[4] shell and IoT-consistent device banners on honeypots that masqueraded as vulnerable IoT devices in an effort to collect Mirai binaries. All incoming Telnet traffic was logged and all binaries downloaded via `wget` or `tftp` were saved for analysis[5]. Combined with binaries collected by Akamai, Krebs' CDN, and those uploaded to VirusTotal, this approach netted researchers 1,028 unique Mirai samples.

Of these samples, 74% were for MIPS 32-bit, ARM 32-bit and x86 32-bit architectures — common architectures for embedded and IoT devices. From the samples, researchers were able to extract 48 distinct IP blacklists, username-password dictionaries, and 67 distinct C2 domains. Of interest is that contained in the Mirai blacklist — that is, the list of IP addresses that the botnet does *not* attempt to infect or attack — was the United States' Department of Defense [13].

From the botnet's interaction with the researchers' honeypots, Antonakakis et al. also observed that some versions of the malware attempt to "conceal its presence by deleting the downloaded binary and obfuscating its process name in a pseudorandom alphanumeric string". This results in Mirai not persisting across reboots since the malicious code is stored only in RAM. Additionally, in order to fortify itself, Mirai has been shown to kill other competing processes (such as those bound to TCP ports 23 and 2323), as well as other strains of malware on the system [13].

5 Evolution of the Malware and the Release of its Source Code

Between the attack on security blog *Krebs on Security* and that on Dyn, Mirai's source code was published online (on Friday, September 30th 2016 at 19:50:52 UTC [22]). This resulted in an "explosion" of the Mirai ecosystem as other variants of the malware were created [13].

[4] BusyBox is the most common shell on IoT devices [21].

[5] *Remark:* researchers prevented collateral damage caused by infected honeybots by blocking all other outgoing requests (*i.e.,* scanning and attack traffic).

These were not the botnet's first major evolutionary step, however. As discussed in Section 4.5, huge numbers of unique malware samples were analyzed. Some of the changes that researchers noted from these binaries occurred *before* the release of the source files. In particular, Antonakakis et al. observed that as early as mid-September the malware changed from IP-based C2 lists to domain-based C2s. At about the same time, samples of the malware had also evolved to implement obfuscation techniques such as deleting its binary and attempts to hide its process ID. Samples collected on September 29th — the day before the source code was made public — even included functionality to aggressively kill competing malware on infected devices.

Of course, given access to the source code, a number of programmers created variants of the malware which tackled other issues. One of the most prominent and successful strains that appeared after the original source code was published was the variant that targeted CWMP devices (see Section 4.3 on page 6).

5.1 How NAT Limits Possible Victims

To compromise non-Internet-facing devices, Mirai originally relied on services like Telnet being exposed to the Internet through port forwarding, UPnP, etc. This is because network address translation (NAT) — an OSI Layer 3 service that nearly all IPv4 routers provide — makes private IP addresses (*i.e.,* those in RFC-1918: `10.0.0.0/8`, `172.16.0.0/12`, and `192.168.0.0/16`) not routable from the Internet, and hence not accessible without specific exceptions like a port forward on TCP port 23, for example.

Due to the nature of layer three IP routing protocols, devices with NATed or private IP addresses are unable to "spoof" or forge their IP address when communicating with other hosts on the Internet. This has relevance to denial of service attacks (and in particular to bandwidth depletion DoS attacks) because of DNS flood techniques called DNS reflection and DNS amplification.

5.2 DNS Reflective and Amplification Attacks

While typical DNS flood DoS attacks are designed to overwhelm a particular authoritative nameserver by requesting huge numbers of nonexistent subdomains such as `s34i8-efrm.victim.com`, DNS reflection utilizes a spoofed source IP address to cause legitimate nameservers to send the query response to a host different than the one that made the query request. Informally, this may be a device `1.2.3.4` making a DNS query while claiming to be `5.6.7.8`. The nameserver that handles the request will send the response to the host that it believes made the request: `5.6.7.8` [23]. As UDP is a connectionless protocol, detection of such address forgery is very difficult to detect.

On its own, however, DNS reflection is not inherently dangerous. The destructive potential of DNS reflection emerges only when combined with DNS amplification [24]. DNS amplification uses DNS extensions such as EDNS0 and DNSSEC to cause the server being duped by the spoofed IP address to respond with exceptionally large messages. As is noted in [23], "EDNS0 allows the DNS response to be larger than the original 512 allowed". Since DNSSEC is becoming more and more popular (and requires more data to be transmitted), more and more servers are supporting EDNS0 and are therefore capable of sending responses of up to 4,096 bytes. Attackers may then study legitimate DNS records to determine which ones contain the most data, and also use DNSSEC for the increased payload containing cryptographic data. Through such means, it is possible for malicious DNS query responses to reach up to 100× the size of the original request [23].

This attack vector made creation of a variant of Mirai that infected Internet-facing devices lucrative. This threat became a reality when the RCE exploit in a SOAP configuration endpoint was found to compromise the CWMP protocol and consequently the botnet was able to infect routers and modems [13].

6 Lessons

When considering the large-scale fallout that Mirai caused on global IT infrastructure, it is easy to overlook the fact that the malware performed no real "hacking", in the typical sense of the word; there was no zero-day exploit, no large-scale brute force attack, nor a hidden backdoor in the infected devices that allowed the malware to spread as it did. Rather, Mirai scanned TCP ports 23 and 2323 — used by the antiquated and insecure Telnet protocol — for hosts that were online, and then attempted ten pseudorandom username-password pairs from a list of default administrative credentials for popular IoT devices. This is to say, the unsophisticated dictionary attacks that allowed Mirai to temporarily cripple the Internet were preventable if either of the following had happened:

(a) default administrative credentials changed to have more secure passwords,

(b) unused legacy protocols such as Telnet disabled on networked devices,

Additionally, even if defunct protocols and default passwords were left untouched, the attacks were even further preventable if either of the following had been implemented:

(a) firewalls rules to filter egress Telnet traffic, thus stopping an infected device from scanning for other devices, or

(b) firewalls rules to filter ingress Telnet traffic, preventing hosts on the Internet from contacting vulnerable devices at all.

6.1 Lessons for Network Administrators

In addition to the important lesson that network administrators must pay as much attention to egress network traffic as ingress traffic — an elementary yet oft-forgotten principle — the Mirai attack has also demonstrated the importance of network administrators discerning whether their application requires a recursive or non-recursive DNS server.

In most cases, enterprise-level networks configure public-facing DNS servers because these servers are the authoritative nameservers for the organization's domain. Such nameservers do *not* need to be open resolvers. That is, if queried for a domain that does not lie within the zone(s) it controls, the server should respond with a `NXDOMAIN` failure rather than passing the query to some preconfigured forwarder or looking at root hints [25]. This prevents an organization's nameservers from participating in an attack on another authoritative nameserver when queried for bogus domains.

6.2 Beyond Password Security

Cybersecurity expert Robert Graham points out in [18] that while it is easy to cast blame on improper use of authentication as a security mechanism, there are other more complex lessons to observe from the Mirai attacks. One such point is that placing all blame on use of default passwords demonstrates a misunderstanding for how IoT devices work and how they are deployed [26].

Mirai infections specifically targeted IP cameras. Such devices are "placed at remote sites miles away, up on the second story where people can't mess with them. In order to reset them, you need to put a ladder in your truck and drive 30 minutes out to the site, then climb the ladder (an inherently dangerous activity)" [26]. Instead, resets are done over Telnet using a hard-coded password. Graham claims that some people see use of Telnet and a hard-coded password and "make assumptions." Rather, fixing the password issue (as some reports have recommended) would simply "mean the manufacturer would create a different, custom backdoor that hackers would eventually reverse engineer, creating MiraiV2 botnet." Graham instead suggests that the real solution is to engineer a better standard for remote resets that does not involve the use of vulnerable protocols [26, 27].

Later, in a 2018 article responding to government efforts to improve IoT cybersecurity by forcing automatic updates as soon as a vulnerability or attack like Mirai is discovered, Graham made another claim: "naïve solutions to the manual patching problem, like forcing auto-updates from vendors, increase rather than decrease the danger" [28]. He went on to say the following:

> Manual patches that don't get applied cause a small, but manageable constant hacking problem. Automatic patching causes rarer, but more catastrophic events when hackers hack the vendor and push out a bad patch. People are afraid of Mirai, a comparatively minor event that led to a quick cleansing of vulnerable devices from the Internet. They should be more afraid of notPetya, the most catastrophic event yet on the Internet that was launched by subverting an automated patch of accounting software.
>
> [Vulnerabilities] aren't even the problem. Mirai didn't happen because of accidental bugs, but because of conscious design decisions. Security cameras have unique requirements of being exposed to the Internet and needing a remote factory reset, leading to the worm. [28]

Graham's writing speaks of a less common belief that the problem that allowed Mirai to cripple the Internet was not in failing to protect against a particular thread model, but that the world had been operating on a flawed threat model and had ignored more important issues [28].

7 Conclusion

The Mirai botnet relied on recent growth of IoT and the average person's dissonance with secure passwords and, in general, digital security to create a network of compromised IP cameras, DVRs, printers, and other small networked devices. Collectively, using hundreds of thousands of compromised devices, Mirai was able to temporarily paralyze some of the world's top content delivery networks by generating DDoS traffic of up to 623Gbps.

After the source code's release, variants of Mirai appeared that

targeted Internet-facing devices such as consumer routers. This enabled attacks involving spoofed IP addresses resulting in payloads of up to 100× the amount of traffic sent by the botnet being delivered to the victim through DNS reflection and amplification attacks.

Through these events, we have observed the importance of seemingly small preventative measures as discussed in Section 6. That is, in short, that default settings can cause huge fallout, that passwords do indeed matter, and that unused protocols and network ports should be disabled — especially if they expose outdated and insecure services.

A consequence of the damage caused by Mirai should not be that consumers and industry shy away from IoT, rather that we keep security in mind as we connect more and more devices to the Internet. The Internet of Things has potential to bring great advances to the daily life of millions of people, but only if it is done properly and securely.

References

[1] *IDC Forecasts Worldwide Spending on the Internet of Things to Reach $745 Billion in 2019, Led by the Manufacturing, Consumer, Transportation, and Utilities Sectors*, Jan. 2019. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS44596319.

[2] *IDC Forecasts Worldwide Technology Spending on the Internet of Things to Reach $1.2 Trillion in 2022*, Jun. 2018. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS43994118.

[3] Z. Ma, M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, *USENIX Security '17 - Understanding the Mirai Botnet*, Sep. 2017. [Online]. Available: https://www.youtube.com/watch?v=1pywzRTJDaY.

[4] *Internet of Things and the Rise of 300 Gbps DDoS Attacks*, Feb. 2017. [Online]. Available: https://www.akamai.com/us/en/multimedia/documents/social/q4-state-of-the-internet-security-spotlight-iot-rise-of-300-gbp-ddos-attacks.pdf.

[5] B. Knieriem, X. Zhang, P. Levine, F. Breitinger, and I. Baggili, "An Overview of the Usage of Default Passwords," in *Digital Forensics and Cyber Crime*, P. Matoušek and M. Schmiedecker, Eds., Cham: Springer International Publishing, 2018, pp. 195–203, ISBN: 978-3-319-73697-6.

[6] D. Fraunholz, D. Krohmer, S. Duque Antón, and H. D. Schotten, "Investigation of Cyber Crime Conducted by Abusing Weak or Default Passwords with a Medium Interaction Honeypot," in *International Conference On Cyber Security And Protection Of Digital Services. Cyber Science (Cyber Security-17), International Conference On Cyber Security And Protection Of Digital Services, June 19-20, London, United Kingdom*, IEEE, 2017.

[7] P. G. Inglesant and M. A. Sasse, "The True Cost of Unusable Password Policies: Password Use in the Wild," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10, Atlanta, Georgia, USA: ACM, 2010, pp. 383–392, ISBN: 978-1-60558-929-9. [Online]. Available: http://doi.acm.org/10.1145/1753326.1753384.

[8] U. D. of Health, *Data Breach Expands to Include More Victims*, Apr. 2012. [Online]. Available: http://udohnews.blogspot.com/2012/04/data-breach-expands-to-include-more.html.

[9] O. of Inspector General, O. of Audits, and Inspections, *Audit Report: Management of Bonneville Power Administration's Information Technology Program*, Mar. 2012. [Online]. Available: https://www.bpa.gov/news/pubs/Audits/audit-2012-IT-IG-Final-Report.pdf.

[10] J. Vijayan, *Weak passwords still the downfall of enterprise security*, Apr. 2012. [Online]. Available: https://www.computerworld.com/article/2503105/weak-passwords-still-the-downfall-of-enterprise-security.html.

[11] B. Krebs, *Europe to Push New Security Rules Amid IoT Mess*, Oct. 2016. [Online]. Available: https://krebsonsecurity.com/tag/xc3511/.

[12] G. M. Graff, *How a Dorm Room Minecraft Scam Brought Down the Internet*, Dec. 2017. [Online]. Available: https://www.wired.com/story/mirai-botnet-minecraft-scam-brought-down-the-internet/.

[13] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the Mirai Botnet," in *26th USENIX Security Symposium (USENIX Security 17)*, Vancouver, BC: USENIX Association, 2017, pp. 1093–1110, ISBN: 978-1-931971-40-9. [Online]. Available: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis.

[14] D. FitzGerald and R. McMillan, *Cyberattack Knocks Out Access to Websites*, Oct. 2016. [Online]. Available: https://www.wsj.com/articles/denial-of-service-web-attack-affects-amazon-twitter-others-1477056080.

[15] S. Hilton, *Dyn Analysis Summary Of Friday October 21 Attack*, Oct. 2016. [Online]. Available: https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/.

[16] R. A. Hallman, J. Bryan., G. Palavicini., J. Divita., and J. Romero-Mariona., "IoDDoS — The Internet of Distributed Denial of Service Attacks: A Case Study of the Mirai Malware and IoT-Based Botnets," in *Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security - Volume 1: IoTBDS,*,

INSTICC, SciTePress, 2017, pp. 47–58, ISBN: 978-989-758-245-5.

[17] N. Woolf, *DDoS attack that disrupted internet was largest of its kind in history, experts say*, Oct. 2016. [Online]. Available: https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet.

[18] R. Graham, *Some notes on today's DNS DDoS*, Oct. 2016. [Online]. Available: https://blog.erratasec.com/2016/10/some-notes-on-todays-dns-ddos.html#.XU0YQmiYVhE.

[19] *Discovering Mirai-infected IoT devices via Merit's network telescope*, May 2017. [Online]. Available: https://www.merit.edu/discovering-mirai-infected-iot-devices-via-merits-network-telescope/.

[20] M. Bailey, E. Cooke, F. Jahanian, and J. Nazario, "The Internet Motion Sensor: A Distributed Blackhole Monitoring System," Jan. 2005.

[21] R. Graham, *RSAC 2017 - Robert Graham on Mirai and IoT Botnet Analysis*, Mar. 2017. [Online]. Available: https://archive.org/details/RSAC2017RobertGraham.

[22] Jgamblin, *jgamblin/Mirai-Source-Code*. [Online]. Available: https://github.com/jgamblin/Mirai-Source-Code/blob/master/ForumPost.md.

[23] L. Rozen, *DNS Reflective Attacks*, Oct. 2017. [Online]. Available: https://blog.radware.com/security/2016/12/dns-reflective-attacks/.

[24] *DNS Amplification DDoS Attacks*. [Online]. Available: https://www.cloudflare.com/learning/ddos/dns-amplification-ddos-attack/.

[25] AT&T Tech Channel, *Protecting DNS Servers From Amplification Attacks | AT&T ThreatTraq Bits*, Mar. 2016. [Online]. Available: https://www.youtube.com/watch?v=KvJX9UMTg6g.

[26] R. Graham, *That "Commission on Enhancing Cybersecurity" is absurd*, Dec. 2016. [Online]. Available: https://blog.erratasec.com/2016/12/that-commission-on-enhancing.html#.XU0YSGiYVhE.

[27] ——, *Notes on the UK IoT cybersec "Code of Practice"*, Oct. 2018. [Online]. Available: https://blog.erratasec.com/2018/10/notes-on-uk-iot-cybersec-code-of.html#.XU0YX2iYVhE.

[28] ——, *Your IoT security concerns are stupid*, Jul. 2018. [Online]. Available: https://blog.erratasec.com/2018/07/your-iot-security-concerns-are-stupid.html#.XU0YVWiYVhE.