Scanning for Vulnerable Devices in the Internet of Things

Conference Paper · October 2015

DOI: 10.1109/IDAACS.2015.7340779

2 authors, including:

George Markowsky
Missouri University of Science and Technology
126 PUBLICATIONS **2,755** CITATIONS

The 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
24-26 September 2015, Warsaw, Poland

Scanning for Vulnerable Devices in the Internet of Things

Linda Markowsky and George Markowsky
UMaine Cybersecurity Laboratory
University of Maine
Orono, Maine, USA

***Abstract*—Many smart, resource-constrained, and seldom-updated devices in the Internet of Things present unanticipated vulnerabilities. The Internet Census 2012 scanned for such devices to construct its Carna Botnet, which then surveyed the entire IPv4 address space. This census provides an order of magnitude for the number of devices vulnerable to just one type of attack. Finally, three scans of different types demonstrate how to scan for vulnerable devices in the Internet of Things. The first uses Shodan to find vulnerable Cayman DSL routers; the second uses Masscan to find devices vulnerable to Heartbleed, and the third used Nmap and PFT to find and connect to vulnerable networked printers.**

Keywords—Internet of Things; IoT; cybersecurity; Internet Census 2012; Shodan; Masscan; Nmap; PFT

I. Introduction: Vulnerabilities of Devices in the Internet of Things (IoT)

A. *Easily-Understood Attacks, Including Physical Attacks, Eavesdropping, and Loss of Data Integrity*

It is estimated that "by 2020, there will be 50 to 100 billion devices connected to the Internet" [1]. "Sensors are expected to be attached to all the objects around us, so these can communicate with each other with minimum human intervention" [1, p. 447].

In such a dense, pervasive computing environment, IoT-connected *things* will be vulnerable to physical attacks, eavesdropping, and loss of data integrity. These common, easily-understood vulnerabilities, as well as more general security and privacy issues, must be mitigated to ensure the public's acceptance and trust. "IoT is a community based approach where the acceptance of the users (e.g. general public) is essential. Therefore, security and privacy protection requirements need to be carefully addressed in order to win the trust of the users" [1, p. 445].

B. *Fragmentation of Large Security Protocol Packets Due to Resource Constraints*

As the Internet evolves into the Internet of Things, many resource-constrained devices, when connected, will communicate using existing protocols. Clearly, "a resource-constrained network that relies on lossy and low-bandwidth channels for communication between small nodes, regarding CPU, memory, and energy budget ... [and] the use of small packets ... may result in the fragmentation of larger packets of security protocols. This may open new attack vectors for state exhaustion DoS attacks, ... e.g., if the fragmentation is caused by large key exchange messages of security protocols" [2]. Furthermore, "the IoT brings communication patterns that are unusual in traditional networks, and thus are not sufficiently supported by end-to-end Internet security protocols" [2, p. 539]. Today, "IoT security solutions are often tailored to the specific scenario requirements without considering interoperability with Internet protocols," and so "the direct use of existing Internet security protocols in the IoT might lead to inefficient or insecure operation" [2, p. 532]. Reliable end-to-end security in the IoT will require mechanisms to prevent the fragmentation of existing protocols by small but smart connected *things* from leading to attacks.

C. *Inability of Things to Use Cryptographic Algorithms Due to Energy Constraints and Lack of Scalability*

Cryptography is essential to ensure data confidentiality, integrity, and authentication, often summarized by the acronym CIA. As more and more sensitive, personal information is transmitted by connected consumer devices and even smart health monitors, ensuring the CIA of such data becomes increasingly important. Many smart, connected *things*, however, "are characterized by low capabilities in terms of both energy and computing resources ... and thus, they cannot implement complex schemes supporting security" [3]. Viewed from the "bottom-up" perspective of a *thing* in the IoT, "cryptography is the cornerstone for network infrastructure protection ... [and] if cryptography is the brick, the mortar is key-management infrastructures." [4]. These current security mechanisms, based on "traditional public-key infrastructures will almost certainly not scale to accommodate the IoT's amalgam of contexts and devices" [4], necessitating the creation of new schemes to ensure data and network protection.

II. The Internet Census 2012 Found More Than 1.6 Million Vulnerable Internet-Connected Things

The Internet Census 2012, while illegally obtained, nevertheless provides an estimate of the number of Internet-connected *things* in 2012 vulnerable to passwordless login or login using default credentials.

The census began unintentionally: "While playing around with the Nmap Sripting Engine (NSE) we discovered an amazing number of open embedded devices on the Internet. Many of them are based on Linux and allow login to standard BusyBox with empty or default credentials. We used these devices to build a distributed port scanner to scan all IPv4 addresses" [5]. The port scanner consisted "of two parts. The first one is a telnet scanner which tries a few different login combinations, e.g. root:root, admin:admin and both without passwords. The second part manages the scanner" [5, Sec. 2].

When run, the somewhat shady researchers found that "the vast majority of all unprotected devices are consumer routers or set-top boxes ... [but also found] IPSec routers, BGP routers, x86 equipment with crypto accelerator cards, industrial control systems, physical door security systems, big Cisco/Juniper equipment" [5, Sec. 3.1]. The researchers found more than 1.6 million devices vulnerable to this simple login credential attack, with vulnerable devices found throughout the world.

About 420 thousand of these vulnerable devices comprised the "Carna Botnet," which then took the census. The vulnerable *things* were capable computers, each powerful enough to run "a modified version of fping [and] ... a modified version of libevents asynchronous reverse DNS sample code" [5, Sec. 5.1 and 5.2]. In addition, "a number of the MIPS machines had enough RAM and computing power to run a downsized version of Nmap," and even some of the smaller devices, not used in the botnet, "provided a few diagnosis commands like ping, and interestingly traceroute on a limited shell. We developed a modified version of our telnet scanner to find and log into these devices" [5, Sec. 5.3 and 5.5].

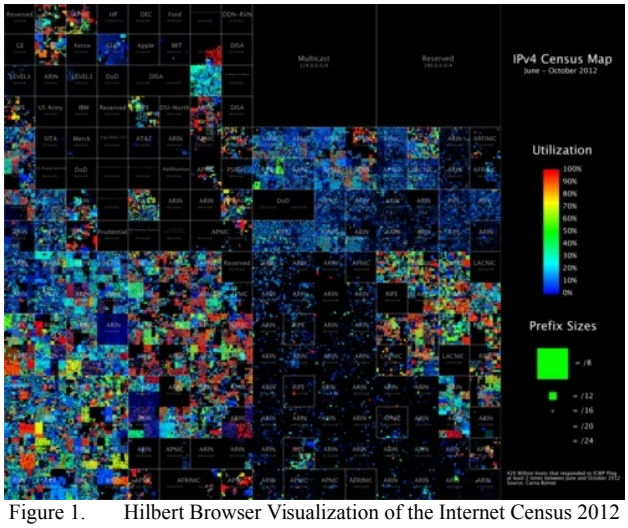

Figure 1. Hilbert Browser Visualization of the Internet Census 2012

The Internet Census researchers provided two visualizations of their results: a modified Hilbert browser [6][7] and a spectacular animated worldwide map [8], shown in Figures 1 and 2, respectively, and made their data public for further research [9].

The researchers concluded: "A lot of devices and services we have seen during our research should never be connected to the public Internet at all. As a rule of thumb, if you believe that 'nobody would connect that to the Internet, really nobody', there are at least 1000 people who did. Whenever you think 'that shouldn't be on the Internet but will probably be found a few times' it's there a few hundred thousand times. Like half a million printers, or a million webcams, or devices that have root as a root password" [5, Sec. 8].

The results of the Internet Census 2012 were validated independently [10], and a new census, to be conducted legally, is now underway [5].

III. Scanning for Vulnerable Devices

In order to demonstrate how to search for vulnerable devices in the IoT, the authors conducted three searches, each sing a different method. The first used Shodan [11], known as an index of banners, to find vulnerable Cayman DSL routers. The second used Masscan to quickly search a large address space for devices vulnerable to the Heartbleed Bug. Finally, the third used Nmap and PFT to find and connect to vulnerable networked printers.

A. *Example 1: Using Shodan to Find Open Cayman DSL Routers*

Unlike search engines such as Google, which index content available on the Internet, Shodan indexes banners returned by devices during the establishment of a connection to an open port. Many routers, printers, cameras, and other devices openly broadcast their existence and even their device type and firmware version number. By correlating known exploits (using an exploit database such as the NIST National Vulnerability Database, Mitre's Common Vulnerabilities and Exposures,

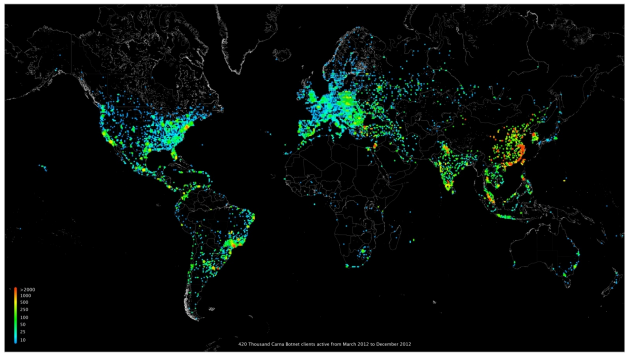
Figure 2. Animated Map Visualization of the Internet Census 2012

or Shodan Exploits [12][13][14]) with the indexed device banners (using Shodan), many vulnerable devices are easily detected.

As an example of this type of search, the authors used Shodan Exploits and Shodan to find routers vulnerable to denial-of-service, or DoS, attacks. This search can be performed anonymously, as it does not require the user to login to a Shodan account.

The search began in Shodan Exploits by entering "router" in the search box and clicking "hardware" under "platform" in the left column. The exploit database returned many router vulnerabilities, including the "Cayman 3220-H DSL Router 1.0/GatorSurf 5.3 DoS Vulnerability," which permits "large usernames or passwords sent to the router's HTTP interface [to] restart the router. [The] router log will show 'restart not in response to admin command'" [15].

Having found a vulnerable type of router, the search continued in the Shodan (banner) database by searching for "3220-H," which returned the IP address and location of many Cayman routers, most of which had open and accessible administrator accounts. These open routers were vulnerable not only because of a known shortcoming in the firmware but also because the open administrator accounts allow anyone, including malicious outsiders, to reconfigure or restart the router and even set a password unknown to the router's owner. That is, no specific exploit code is needed to conduct a DoS attack on these open Cayman routers.

B. Example 2: Using Masscan to Find Devices Vulnerable to Heartbleed

As a second example, the authors scanned for devices in the IoT vulnerable to Heartbleed. Acccording to [16], "the Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. This weakness allows stealing the information protected, under normal conditions, by the SSL/TLS encryption used to secure the Internet. SSL/TLS provides communication security and privacy over the Internet for applications such as web, email, instant messaging (IM) and some virtual private networks (VPNs). The Heartbleed Bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret keys used to identify the service providers and to encrypt the traffic, the names and passwords of the uers and the actual content. This allows attackers to eavesdrop on communications, stead data directly from the services and users and to impersonate services and users" [16]. Clearly, network security administrators have a vested interest in scanning their networks for all attached devices vulnerable to Heartbleed.

Nmap, despite being the best-known and most widely used scanner, is not necessarily the fastest, and so when scanning a large IP address space, other scanners such as Masscan will likely provide faster results [17][18][19]. Masscan uses familiar Nmap-style options but, unlike Nmap, operates asynchronously using a "driver that directly transfers packets to network hardware, bypassing the kernal; a user-mode TCP stack; and mutex-free thread synchronization [17]. While admittedly less detailed and less accurate than Nmap, Masscan is remarkably fast and thus suitable for an initial scan of a large address space.

Using Masscan, the authors scanned (with permission) the entire University of Maine System's Class B wired address space for devices vulnerable to the Heartbleed Bug. The command:

```
#masscan <target IP address space>/16 -p443 -S
<my IP address> --rate 1000 –heartbleed >&
masscan.output.filename
```

returned very quickly and identified twelve *things* vulnerable to Heartbleed. Collecting these twelve addresses in a text file made it easy to scan for more reliable, detailed information using Nmap:

```
#nmap -Pn -A -iL heartbleed.vuln.ip.addresses.txt >&
nmap.output.filename
```

Among the vulnerable devices were Polycom systems, which were state-of-the-art not long ago but were rapidly eclipsed by more convenient software, underscoring the danger posed by infrequently updated but ever-connected "things" in the IoT.

C. Example 3: Using Nmap and PFT to Find and Connect to Vulnerable Printers

As mentioned in the previous section, Masscan is faster but less accurate than Nmap, the de facto standard tool for network security scanning. This is because Masscan, unlike Nmap, is stateless, and "a stateless scanner cannot detect dropped packets in order to retransmit and throttle its send rate. If a busy router half way along the network path drops 80% of the scanner's packet flood, the [stateless] scanner will still consider the run successful and print results that are woefully inaccurate. Nmap, on the other hand, saves extensive state in RAM while it runs ... Nmap marks each probe with sequence numbers, source or destination ports, ID fields, or other aspects ... , which allow it to recognize responses" [20].

Therefore, to follow up on the Masscan results from the previous section, the authors (with permission) conducted a TCP SYN scan limited to port 443 using Nmap:

```
#nmap -sS -p443 <ums network>/16 >&
output.filename
```

The results revealed a surprising number of networked printers, many with open, passwordless admin web interfaces. Networked printers are, in reality, smart *things* or specialized computers and may be vulnerable to a variety of attacks, including document leakage, denial of

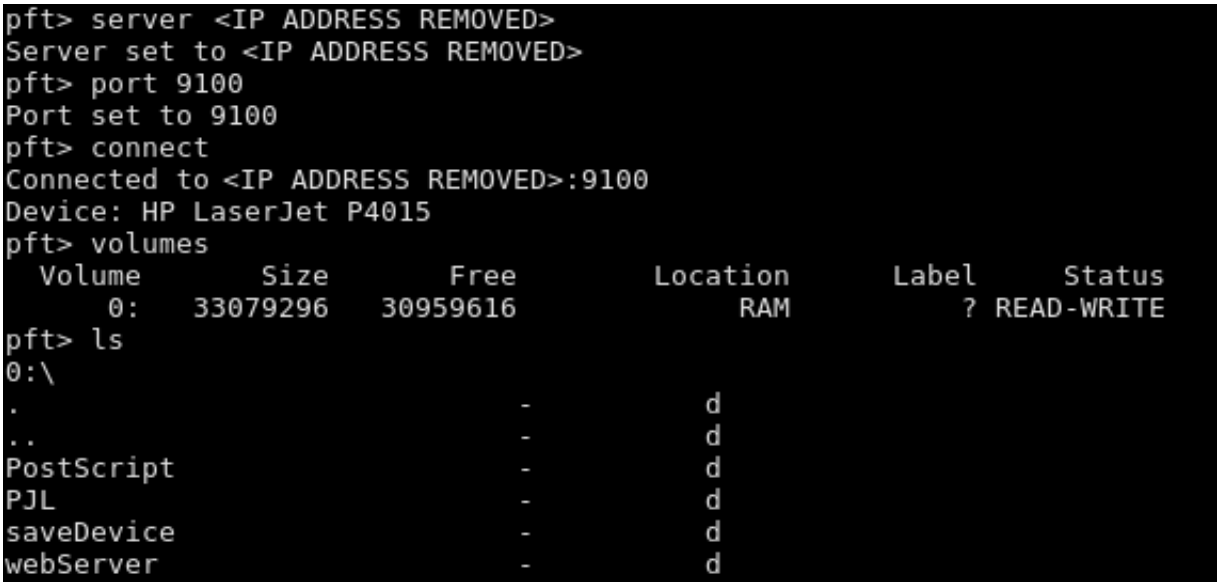

Figure 3. Using PFT to Connect to a Networked Printer

service, and stealth (or zombie) scans [21], by which a hacker uses the printer to mask the true source of a hostile or intrusive Nmap scan.

To explore these networked printers further, the authors used Nmap to scan for common printer ports, saving the results in both XML and text formats:

#nmap -p 9100, 515, 631 <ums network>/16 -oX printer-scan-results.xml >& printer-scan-results.txt

The scanned ports, which are commonly used by printers and print servers, are:

- 9100 = the RAW port for most printers, also known as the direct-IP port;
- 515 = the LPR/LPD port for most printers, as well as older print servers;
- 631 = the IPP port for most modern printers and CUPS-based print servers [22].

The scan showed that of the 65536 IP addresses in the UMS class B network, 6565 hosts responded, and of these:

- 620 hosts reported port 9100/tcp "open"
- 664 hosts reported port 515/tcp "open"
- 509 hosts reported port 631/tcp "open"

That is, about 10% of the address space was used by hosts that responded, and of these, about 10% appeared to be printers with hackable ports. Many of the printers voluntarily leaked valuable information by means of their names that revealed the printer's location and/or model number. One vulnerable printer appeared to be located in the President's office on one of the UMS campuses, where sensitive documents might well be printed. Document leakage from such a printer could prove damaging to the University.

Once an open networked printer is located, connecting to port 9100 is relatively easy using the PFT tool [23][24]. The results of one such connection are shown in Figure 3.

IV. Conclusion

As the IoT becomes a reality, many smart "things" are likely to be connected and then forgotten. Many may seldom, if ever, be updated and will provide a foothold for hackers into networks around the world. Scanning for vulnerable devices in the IoT is imperative in order to ensure adequate security and privacy.

References

[1] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: a survey," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, p. 417, First Quarter 2014.

[2] T. Heer, O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. Kumar, K. Wehrle, "Security challenges in the IP-based internet of things," Wireless Pers Commun, vol 61, p. 533, September 2011.

[3] L. Atzori, A. Iera, and G. Morabito, "The internet of things: a survey," Computer Networks, vol. 54, p. 2801, 2010.

[4] R. Roman, P. Najera, and J. Lopez, "Securing the internet of things," IEEE Computer, p. 53, September 2011.

[5] Carna Botnet, "Internet census 2012: Port scanning /0 using insecure embedded devices," Abstract [Online]. Available: http://internetcensus2012.bitbucket.org/paper.html

[6] ISI (Information Sciences Institute), "ANT censuses of the Internet address space," [Online]. Available: http://www.isi.edu/ant/address/

[7] Carna Botnet, "Internet census 2012: Hilbert browser," [Online]. Available: http://internetcensus2012.bitbucket.org/hilbert/index.html

[8] Carna Botnet, "Internet census 2012: geovideo.gif," [Online]. Available: http://internetcensus2012.bitbucket.org/images/geovideo.gif

[9] Carna Botnet, "Internet census 2012: Download," [Online]. Available: http://internetcensus2012.bitbucket.org/download.html

[10] H. C. Maan, J. Santanna, A. Sperotto, P. T. de Boer, “An approach to validate the Internet census 2012,” B.Sc. report, University of Twente, Enschede, The Netherlands, 2014 [Online]. Available: http://www.utwente.nl/ewi/dacs/assignments/completed/bachelor/reports/2014_B.Sc_Assignment_D.Maan.pdf
[11] J. Matherly, Shodan, 2015 [Online]. Available: http://www.shodanhq.com/
[12] NIST (National Institute of Standards and Technology), National Vulnerability Database, 2015 [Online]. Available: http://web.nvd.nist.gov/view/vuln/search
[13] Mitre, CVE (Common Vulnerabilities and Exposures), 2015 [Online]. Available: http://cve.mitre.org/find/index.html
[14] J. Matherly, Shodan Exploits, 2015 [Online]. Available: http://www.shodanhq.com/exploits
[15] Offensive Security, Cayman 3220-H DSL Router 1.0/GatorSurf 5.3 DoS Vulnerability, 2015 [Online]. Available: http://www.exploit-db.com/exploits/19923/
[16] Codenomicon, The Heartbleed Bug, 2014 [Online]. Available: http://heartbleed.com/
[17] R. Graham, “Masscan: the entire Internet in 3 minutes,” 2013 [Online]. Available: http://blog.erratasec.com/2013/09/masscan-entire-internet-in-3-minutes.html#.VPI9EzXh3rc
[18] R. Graham, P. McMillon, and D. Tentler, “Mass scanning the Internet: tips, tricks, results,” [Online]. Available: https://www.defcon.org/images/defcon-22/dc-22-presentations/Graham-McMillan-Tentler/DEFCON-22-Graham-McMillan-Tentler-Masscaning-the-Internet.pdf (slides) and http://www.youtube.com/watch?v=UOWexFaRylM (video)
[19] R. Graham, “MASSCAN: Mass IP port scanner,” [Online]. Available: https://github.com/robertdavidgraham/masscan
[20] G. Lyon, Nmap Network Scanning, Sunnyvale, CA: Insecure.com, LLC, 2008, p. 129.
[21] A. Crenshaw, “Hacking network printers,” 2015 [Online]. Available: http://www.irongeek.com/i.php?page=security/networkprinterhacking
[22] Serverfault, “Find printers with Nmap,” 2013 [Online]. Available: http://serverfault.com/questions/154650/find-printers-with-nmap
[23] Phenoelit, Gray Hat Tools: PFT & Hijetter, Printer exploration, [Online]. Available: http://www.phenoelit.org/fr/tools.html
[24] R. Weiss, “Hacking HP printers for fun & profit,” 2013 [Online]. Available: https://www.altamiracorp.com/blog/employee-posts/hacking-hp-printers-for-fun-profit