Practical Survey on Internet of Things (IoT) Digital Forensics

Chanyang Shin, Prerit Chandok, Ran Liu, Seth Nielson, Timothy Leschke
Whiting School of Engineering, Security Informatics
Johns Hopkins University
Baltimore, United States

Abstract*— The explosion of Internet of Things (IoT) devices in the marketplace provides increased opportunities for a Digital Forensics Investigator to collect useful evidence. However, IoT specific design and operational characteristics present challenges during the various phases of the investigative process. In an effort to overcome some of these obstacles, academia and the forensics community proposed several models for tackling the end to end procedure of investigating IoT devices. However, most of the research are still very conceptual, proactive in nature [1] and focus on one specific evidence collection technique. Moreover, there is little postulation as to what physical information and investigator can be derived from interrogating IoT's logical data. In support of meeting this research need, this paper presents a survey of current IoT data collection methods at the device and network level. It also describes the potential forensically driven information that hinges on the cyber-physical implied data, and proposals for future work in collecting forensic data at the host and network levels.

Keywords—*IoT Forensics, Digital Forensic Investigation, IoT Security, Cyber-physical, Amazon Echo, Echo Dot, Z-wave, HAN, Home Routers.*

I. Introduction

The Internet of Things (IoT) is the interconnection of electronic devices that leverages existing network infrastructure (i.e. the Internet) to integrate the cyber realm with the physical realm. In the context of home automation, examples include the Amazon Echo and Google Nest devices. These devices capture, process, and relay consumer's physical interaction via the logical network. The IoT essentially is underlying foundation that allows a cyber-physical system to operate in an cooperative way.

A. Challenges of IoT Forensics

While many in the past have professed the need for formalizing, designing, and implementing a forensics framework for IoT, a murder case in Arkansas raised the issue to the forefront [2]. In 2017, a man was accused of killing a former police officer, and in support of the investigation, the prosecutor requested Amazon to release voice recordings from the defendant's Amazon Echo. Since Echo stores all of its recordings on Amazon's servers [3], the investigators had to make a formal request to the company for the forensic data. After weeks of legal battle, and only after the defendant gave permission, Amazon agreed to release the records to the prosecutor. In the same case, investigators analyzed the defendant's smart water heater, and noted "exorbitant amount of water being used," which they believe was as result of the defendant's attempt to hose down the crime scene. This case highlights both the potential and complexities that IoT presents for a Digital Forensics Investigator (DFI); end to end encryption, cloud based storage, and legal/privacy considerations.

B. Focusing on the Reactive Process

As defined in Kebande and Ray's "A generic digital forensics investigation framework for Internet of Things (IoT)," Reactive Process occurs after an incident is identified, and is the process of digital forensics. We chose to concentrate our survey on methods that a DFI might employ post incident identification. As with the Amazon Echo scenario above, most cases do not offer law enforcement agencies the luxuries of having pre-arranged agreements with IoT providers to release cloud based information or a pre-deployed data collection system, both which are in-line with the Proactive Process [4]. More often than not, DFIs are given a device after an incident, and they must first figure out what data is available, and how to forensically acquire them. Although there has been some research on cloud forensics, due to legal and privacy issues, DFIs are typically limited to network and device data that are physically accessible.

C. Main Contributions

- Present a survey of network and device level forensics.
- Describe the potential forensically implied data.
- Propose potential research opportunities.

II. Amazon Echo

The Amazon Echo is a smart hub device that allows people to connect other smart devices to it and use voice commands to control it [5]. The Amazon Echo connects to Alexa, a cloud based voice service, that allows people to use voice commands to request the device to perform multiple things starting from answering some basic questions like weather report and daily news to controlling the lights and speakers in your house. For

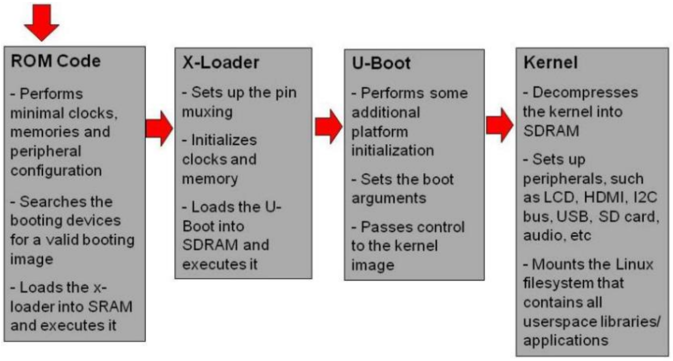

ROM Code
- Performs minimal clocks, memories and peripheral configuration
- Searches the booting devices for a valid booting image
- Loads the x-loader into SRAM and executes it
X-Loader
- Sets up the pin muxing
- Initializes clocks and memory
- Loads the U-Boot into SDRAM and executes it
U-Boot
- Performs some additional platform initialization
- Sets the boot arguments
- Passes control to the kernel image
Kernel
- Decompresses the kernel into SDRAM
- Sets up peripherals, such as LCD, HDMI, I2C bus, USB, SD card, audio, etc
- Mounts the Linux filesystem that contains all userspace libraries/ applications

Fig. 1. The Echo boot sequence: when a system boots, the CPU calls the reset vector to start the code at a location in ROM [6].

our survey, we wanted to analyze the default configurations of the device, obtain firmware information through various methods, analyze the firmware and determine potential vulnerabilities, and finally validate the exploitation techniques we found as potential vulnerabilities on the device. We were able to find the complete breakdown of the Amazon Echo device.

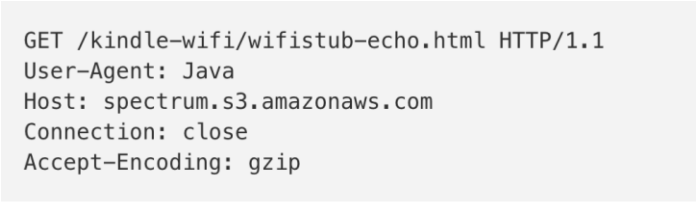
A. Echo Boot Process

To start with, the device is setup after the code is flashed to the Echo ROM, and it look out for a boot device. The echo boots from an external SD card first and if no SD card is found then it fails-over to the internal embedded MMC. After the selection of the boot device, it searches for a file MLO in the first available FAT32 partition.

Figure 1 is basically the TI X-loader program (Texas Instrument) that initializes memory and clocks, and loads Universal-Boot into SDRAM and simply executes it. Universal-Boot is a primary bootloader that sets the boot argument, locates the kernel, and passes control to the kernel image. Now, once the kernel is booted, the peripherals and storage devices, etc. are also loaded. All the Alexa initialization scripts are usually run at this point.

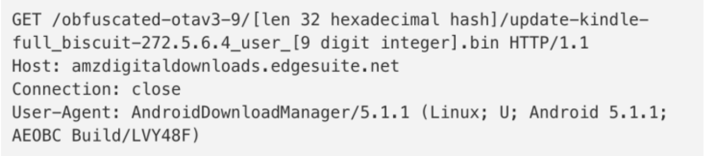
B. Echo Data Extraction

Like many IoT devices, Amazon Echo uses a very simple web server for device setup. The device can be used with the help of a phone app and connects to the home Wi-Fi by emitting signals. The device then further acquires an IP address via DHCP, and can then be connected to the internet normally. Now, this device has Amazon Fire Operating System (Android 5.1.1) in which the updates are pushed as .bin files over HTTP. This makes it possible to intercept over-the-air firmware from Amazon's servers. The .bin files has huge number of APK's and binaries to reverse. To understand Echo, a tested environment should be present to wiretap the echo with iptables and DD-WRT. With this router, we can use iptables to port mirror all traffic on its subnet. However, most of the services that talk over HTTP use TLS connection, that is, data sent to Amazon is encrypted. To intercept the TLS connection with an HTTP proxy by using a browser to navigate to http://alexa.amazon.com, a trusted certificate authority must be installed on the device. As we can see figure2, not all the communication from Echo sent over HTTPS [7].



```
GET /kindle-wifi/wifistub-echo.html HTTP/1.1
User-Agent: Java
Host: spectrum.s3.amazonaws.com
Connection: close
Accept-Encoding: gzip
```

Fig. 2. This shows that the DOT calls itself Kindle and communicates with HTTP.



```
GET /obfuscated-otav3-9/[len 32 hexadecimal hash]/update-kindle-
full_biscuit-272.5.6.4_user_[9 digit integer].bin HTTP/1.1
Host: amzdigitaldownloads.edgesuite.net
Connection: close
User-Agent: AndroidDownloadManager/5.1.1 (Linux; U; Android 5.1.1;
AEOBC Build/LVY48F)
```

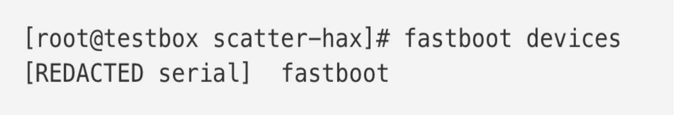
Fig. 3. This figure shows the get HTTP request of obfuscated firmware by the echo device.

Figure 2 above shows the get request of the firmware update and we can easily keep a local copy of this .bin firmware dump from the network. At this point, we can also see that Echo uses Android 5.1.1 which means that Android vulnerabilities are applicable to Amazon Echo devices. The firmware release has build.prop that tells us information about its CPU Mediatek MT8163.

In the burp suite, Fig 3, we could see obfuscated URI as .bin file is not encrypted in any way. We could get the bin file by typing an easy wget command. Hence, we can easily see a typical OTA update filesystem. In this, we have a series of APK's that can be readily decompiled with jadx (Dex to java-decompiler). Some of the interesting things like, alexad, debuggerd, wifid, controld, spotifyd, and some *nix utilities are present in the /bin directory. The /local/models directory in Echo contains British English, American, and German training models after triggering keywords like, 'Alexa', 'Echo', and 'Amazon'. If an attacker wants to hack and change Alexa's voice, then he can accomplish it by changing the Echo's mp3 files in /local/share/earcon directory [8].

C. Determining Forensic Value of Echo

In the reactive forensic analysis, the investigator will identify as many different attack surfaces as possible for that particular device. Now, for the investigator the valuable information to extract from echo could be the things that was searched by the user after triggering the keyword, the user who was looking for the answers, all the metadata like, time of search, IP address of the device, Internet URI and even the things purchased from Amazon. For investigators, there are three main approaches for accessing the device or collecting the valuable data.



```
[root@testbox scatter-hax]# fastboot devices
[REDACTED serial]  fastboot
```

Fig. 4. It shows the access to the bootloader

1. *SD card pinout:* The Citadel guys took advantage of the SD card pinout by configuring an SD card that the Echo could boot from. They made a free account on the TI's website and installed the DM37x SDK on the Ubuntu machine. Then they used the mk3PartSDCard.sh script (from TI wiki) to format the card. Then the MLO, Universal-Boot.bin, and uImage files are moved over to the boot partition, and then the linux filesystem was copied to the ext3 partition labeled 'rootfs'. After accomplishing all this, the investigators SD card will be ready to be booted from by the Amazon Echo and can extract the valuable information then.

2. *Embedded MMC style root:* The Citadel guys were able to get the root privileges through embedded MMC (Multimedia card) debug pins by taking the Echo apart and making an eMMC converter. Also, it is easy to access the locked bootloader by holding the dot key (uber) until the LED light turns green. Fig 4 shows running fastboot devices and gaining access to the bootloader [9].

3. *Joint Test Action Group Pinout:* This hasn't been performed by researchers till now, however it is a promising avenue as it could allow full control of the device. In Fig. 5, JTAG pinout of Echo could allow access to debugging and programming features on the device.

D. *Future Work*

We know that these approaches would allow investigators to access into the filesystem of the Echo. With this further access, the investigators can reverse engineer binaries for vulnerabilities and scan Echo for any hardcoded credentials. After following these approaches, it could also prove that the Echo is a robust and secure device. Hence, further analysis is required to make any judgements [10].

III. Smart Grid Enabled Home Area Networks (HANs)

A Smart Grid is an intelligent monitoring system which delivers electricity from suppliers to consumers and keeps track of all electricity flowing in the system by overlaying the electricity distribution grid with an information and net metering system. The main purpose of the Smart Grid is to control the appliances at consumers' homes to save energy, reduce cost and increase reliability and transparency. A distributed architecture for the HAN (Home Area Network) consists of the smart appliances (AC, refrigerator, etc.), the smart meter, the gateway and a user interface (UI) which is either directly or remotely connected to the gateway. In this architecture, smart appliances (SAs) directly communicate with the gateway and they either directly or indirectly (via the gateway) communicate with the smart meter (SM) using a HAN protocol such as Z-Wave Protocol[11].

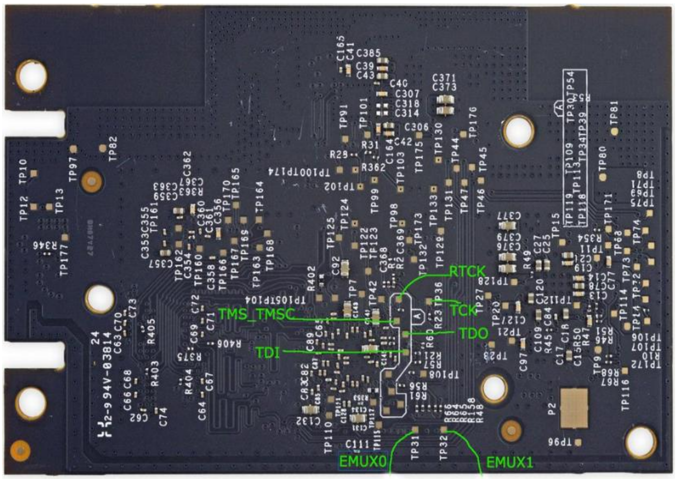


Fig. 5. It is the potential Echo JTAG pinout

A. *Forensic Value of HAN*

One of the most popular HAN protocol, Wi-Fi, can be found in many houses. Once customers want to add a new device to the HAN, the logged data, that logging network related activities in the form of network traffic, user activities can be used to forensics analyze. What's more, the devices information can be used to determine that a person of interest was present at a crime scene or not. Another popular HAN protocol, INSTEON[12], is widely used in the light and security system of house. The authentication process of INSTEON is either through pressing button or sending message to devices, which means the figure print, image and message can be used to forensics analyze. Besides, if one entered in the house, the light will automatically work. The change of data in electricity meter box can be used to determine if someone was in the house at the specific time.

B. *Future Work*

A new survey of HAN can be used to forensics analyze. There are more HAN protocols, such as Z-wave, which is mostly used for remote controls and Zig-bees. Some new devices using Z - Wave protocol such as Shock Sensor that detects the vibrations made by an intruder trying to break into a window or door and sends a Z-Wave® trigger signal to the network, can use data to forensics analyze, we can try to figure out if data stored in such devices can be used to forensics analyze.

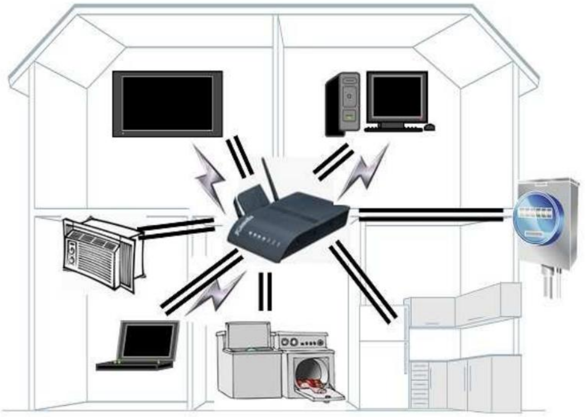


Fig. 6. Distributed architecture for the HAN.

IV. Home Routers

For most home IoT ecosystems, routers serve as the single point of local area network interconnection, and the gateway out to the Internet. Highlighting its importance, many researchers designate routers as a key piece of evidence during an IoT forensic investigation. Oriwoh and Sant [13] described a forensic model that divides the digital crime footprint into three zones. In this context, a home router effectively sits in Zone 2 (gateway and boundary services). Additionally, Perumal, Norwawai, and Raman [14] proposed an alternate forensic model that places a home router as Data Accumulation Platform who evidence is fragile in nature (Figure 7).

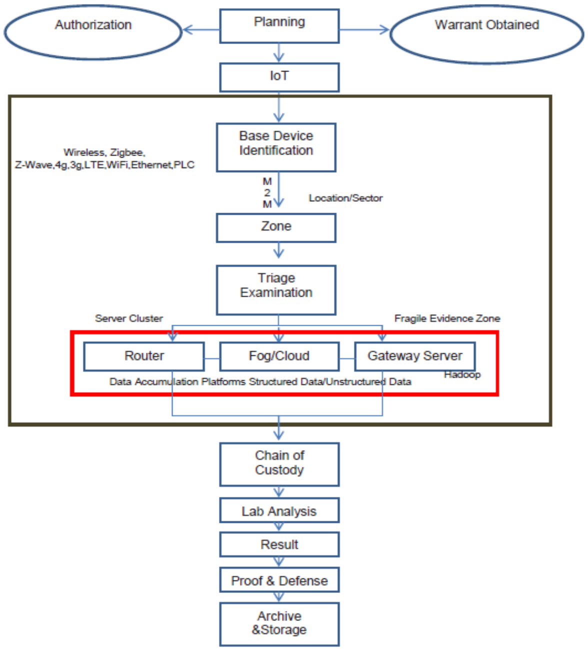


Fig. 7. Perumal, Norwawai, and Raman's IoT Based Digital Forensic Model highlights the importance of a router within an investigative context.

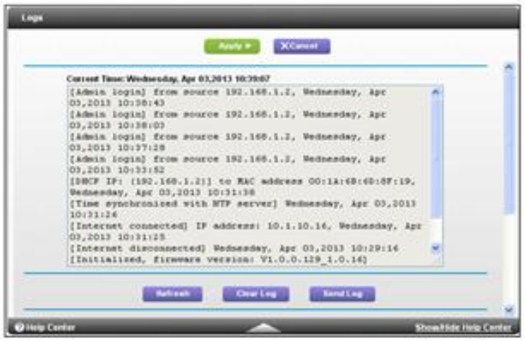
The two characteristics (aggregate and volatile) of data housed on routers make them an attractive source of forensic information, as the devices can store information about all of the IoT devices within the local area network.

A. *Potential for Logging at the Router*

In a survey of 13 brands and 188 different routers [15], Liu, Chen, Yu, and Fu found that majority of home routers perform some sort of logging. In this study, 39.46% of routers logged system activities, which could reveal information on what IoT devices were connected to the local area network. Moreover, small number of routers (6.81%) logged inbound and outbound packets, which can reveal information on network traffic behaviors. The authors noted that many home routers do not log innately, and that their limited storage size hinders usefulness of the logs. However, in current environments, DFIs might be able to capitalize on the grown capabilities of home routers. For example, one of the routers in aforementioned study (Buffalo G300N) only has 32MB and 400MHz single core processor. In contrast, NETGEAR Nighthawk X10 (released in 2016) has 512MB of storage, 1GB of RAM, and runs a 1.7GHz Quad-Core processor [16]. Furthermore, unlike the Buffalo G300N (released in 2010), current NETGEAR's operating system in its default state logs robustly. For example, a standard off-the shelf NETGEAR router will log detailed records of sites visited by connected devices [17]. In the sample log below, we even see the MAC address and IP associations, along with source and destination IP addresses.

B. *Determining Forensic Value of a Router*

System and network logs on a router can assist an investigation in several ways. MAC addresses and IP associations stored on a router can confirm or deny that a person of interest was present at a crime scene. In an IoT setting, a router can serve as a forensic enabler. Although most IoT devices encrypt the contents of their traffic to the cloud, the information regarding the volume data is not obfuscated. Princeton researchers [18] analyzed traffic patterns of several popular IoT devices, to include the Amazon Echo and Google Nest camera. The study revealed that user interaction (i.e. voice command for the Echo or motion sensor activation on the Nest camera) with certain IoT devices creates distinguishable traffic patterns on the wire (Figure 9).



Current Time: Wednesday, Apr 03,2013 10:39:07

```
[Admin login] from source 192.168.1.2, Wednesday, Apr
03,2013 10:38:43
[Admin login] from source 192.168.1.2, Wednesday, Apr
03,2013 10:38:03
[Admin login] from source 192.168.1.2, Wednesday, Apr
03,2013 10:37:28
[Admin login] from source 192.168.1.2, Wednesday, Apr
03,2013 10:33:52
[DHCP IP: (192.168.1.2)] to MAC address 00:1A:6B:6D:8F:19,
Wednesday, Apr 03,2013 10:31:38
[Time synchronized with NTP server] Wednesday, Apr 03,2013
10:31:26
[Internet connected] IP address: 10.1.10.16, Wednesday, Apr
03,2013 10:31:25
[Internet disconnected] Wednesday, Apr 03,2013 10:29:16
[Initialized, firmware version: V1.0.0.129_1.0.16]
```

Fig. 8. Screen shot of default network logs on a NETGEAR Nighthawk router. Logs reveal MAC/IP address associations, as well as timestamped source and destination IPs.

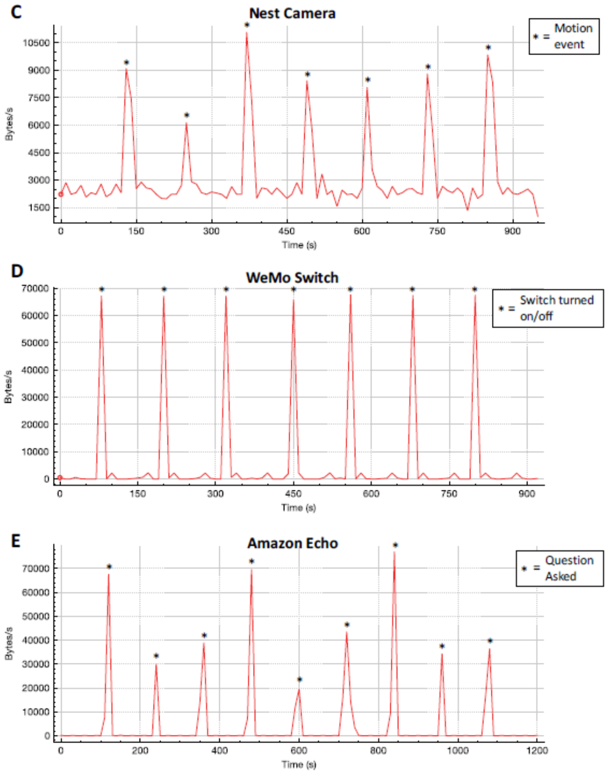


Fig. 9. User interactions results in distinguishable traffic patterns, regardless of the payload being encrypted.

Although not tested in a lab environment, by analyzing the time stamped logs of MAC addresses, IP addresses and packet sizes, router logs can confirm or deny presence of a human being in a house or even certain part of house for a particular timeframe.

C. Future Work.

A new survey of home routers would be useful in determining the usability of currently popular home routers as a forensics enabler. The study should include whether or not the operating system innately performs logging of network traffic, the type of logging it performs (system activity, packets, etc), and the size limit for the log files. We assume that with a 16 fold increase in storage size and over 4 fold increase in processor speed on a router since 2010, the forensic landscape will be more favorable for a DFI today. Once an updated survey is completed and a suitable router is selected, the theory of logging IoT traffic patterns at the router (vs a sniffer located internal to the network as outlined in previous research) should be tested in a lab environment.

Conclusion

A quick search on Compendix for the terms "IoT" and "forensics" only yields 34 published works. The scarcity of journals and papers on this topic is indicative of the emergent nature of IoT forensics. As with most fields in their infancy, early works on IoT forensics have been predominantly theoretical in nature, and aimed to tackle issue such as frameworks and models. This paper is a departure from existing works in that it provides a practical appraisal of forensics methods on popular IoT devices and protocols, and hones in on how the extract data can aid an investigation. Furthermore, the paper consolidates pragmatic research opportunities that will have immediate impact to the art of IoT forensic investigation.

Acknowledgment

We would like to thank Dr. Seth Nielson and Dr. Timothy Leschke of Johns Hopkins University for contributing and continually guiding us in our survey. We would also like to express our gratitude and appreciation to all the individual researchers who helped us in our practical survey of digital forensics of IoT devices.

References

[1] V. R. Kebande and I. Ray, "A generic digital forensic investigation framework for Internet of Things (IoT)," 2016 IEEE 4th International Conference on Future Internet of Things and Cloud, p. 356-362, September 23, 2016.

[2] http://www.cnn.com/2017/03/07/tech/amazon-echo-alexa-bentonville-arkansas-murder-case/)

[3] https://www.cnet.com/news/amazon-echo-teardown-a-smart-speaker-powered-by-amazons-cloud/

[4] S. Alharbi, J. Weber-Jahnke, I. Traore, "The proactive and reactive digital forensics investigation process: a systematic literature review," International Journal of Security and Its Applications, October, 2011.

[5] Amazon.com, Inc, Amazon Echo Product Page, http://www.amazon.com/Amazon-SK705DI-Echo/dp/B00X4WHP5E

[6] OMAPpedia, Bootloader Project, http://omappedia.org/wiki/Bootloader_Project

[7] Asd Amazon Echo Dot System Image, https://blog.padil.la/2017/01/20/amazon-echo-dot-system-image/

[8] Exploring the Amazon Echo Dot, Intercepting firmware updates, https://medium.com/@micaksica/exploring-the-amazon-echo-dot-part-1-intercepting-firmware-updates-c7e0f9408b59#.squqxbj1w

[9] Exploring the Amazon Echo Dot , Into MediaTek utility hell, https://medium.com/@micaksica/exploring-the-amazon-echo-dot-part-2-into-mediatek-utility-hell-b452f62e5e87#.alnvlm5gi

[10] Ike Clinton, Lance Cook, and Dr. Shankar Banik, A Survey of Various Methods for Analyzing the Amazon Echo, Unpublished

[11] Erman Ayday, and Sridhar Rajagopal,"Secure Device Authentication Mechanisms for the Smart Grid-Enabled Home Area Networks,", https://infoscience.epfl.ch/record/188373/files/smart_grid_tech_report.pdf

[12] INSTEON, Developer's guide http://www.insteon.net

[13] E. Oriwoh and P. Sant, "The forensics edge management system: A concept and design," 2013 IEEE 10th International Conference on Autonomic and Trusted Computing, p. 544-550, December 18, 2013.

[14] S. Perumal, N. Norwawi, V. Raman, "Internet of things (IoT) digital forensic investigation model: Top-down forensic approach methodology," 2015 5th International Conference on Digital Information Processing and Communications, p. 19-23, November 9, 2015.

[15] Z. Liu, C. Yu, W. Yu, X. Fu, "Generic network forensic data acquisition from household and small business wirelss routers," 2010 IEEE International Symposium on "A World of Wireless, Mobile and Multimedia Networks, 2010.

[16] https://www.netgear.com/landings/ad7200/?cid=wmt_netgear_organic

[17] https://kb.netgear.com/24224/How-do-I-view-the-activity-logs-of-my-Nighthawk-route

[18] N. Apthorpe, D. Reisman, N. Feamster, "A smart home is no castle: privacy vulnerabilities of encrypted IoT traffic," unpublished.