IoT Security Overview and Attacks

Dylan Dreyer

November 5, 2018

1 Introduction

Due to the development of powerful yet inexpensive hardware and the advancement in software, the Internet of Things (IoT) has become a huge part of today's technologically interconnected world. The IoT is a broad term for the technology that brings people, devices, data, and processes together. An IoT system usually involves a collection of physical devices (vehicles, homes, appliances, etc.) that sense information from the world and communicate over the Internet. Because of the great scope of what the IoT encompasses from physical devices to cloud computing, it is hard to find a general overview, much less a security overview, of what exactly the IoT entails. At the same time however, security is of utmost important in the IoT because IoT devices and associated data directly affect the physical world and us humans. These security concerns range from data privacy threats to physical safety concerns. With the IoT connecting more and more devices to the Internet in the near future, the attack vectors will continue to increase.

2 Security Overview: Smart Home

To demonstrate the vast scope of IoT security threats, let us take an example IoT system of a smart home and examine its vulnerabilities.

2.1 Smart Home Setup

A smart home could consist of many IoT "smart" devices. A short, non exhaustive list of such devices are:

- lights
- security cameras
- locks
- appliances (washer/dryer, refrigerator etc.)
- home controlling devices (Apple HomePod, Amazon Alexa etc.)

The list could go on and on. All of these devices in a smart home can communicate over a network amongst themselves and to the broader Internet. Because of this, they are exposed to a wide variety of network security threats.

2.2 Smart Home Security Threats

Of the many security areas in IoT, there are a few specific areas in which smart home IoT devices are typically targeted. Here are a few examples of vulnerabilities for each security area:

1. **Privacy**
    - A network attacker could observe encrypted network traffic from smart home and use a side channel attack that leaks information about the data [1].
    - An attacker could hack a home controlling device and use its voice listening capabilities to record conversations in the home.
2. **Integrity of operation**
    - An attacker could hack into IoT devices and use them to cause harm to users i.e. tampering with heaters, car controls, lights, security devices etc.
    - An attacker could manipulate the surrounding environment of an IoT device to cause false alarms, send erroneous signals to users, or else send bad data to the Internet. The DolphinAttack, which will be covered in Section 3.1, is an example of this form of attack [4].
3. **Resource usage**
    - An attacker could hack into IoT devices to use them as a compute resource for:
        - running malicious programs and/or bots
        - Denial of Service (DoS) attacks (a real world attack of this kind, the Mirai botnet, will be covered in Section 3.2) [6]
        - cryptocurrency mining (DroidMiner is an example of this type of attack [2])

These are only a few examples of possible IoT security vulnerabilities. As seen in Figure 1, IoT security spans many layers of computer systems and networks. Any of these layers or components can be compromised to form an attack on the IoT as a whole. Though this figure represents Cisco's model of the IoT stack, it serves as a reminder that the IoT is heterogeneous and complex, making it hard to secure.

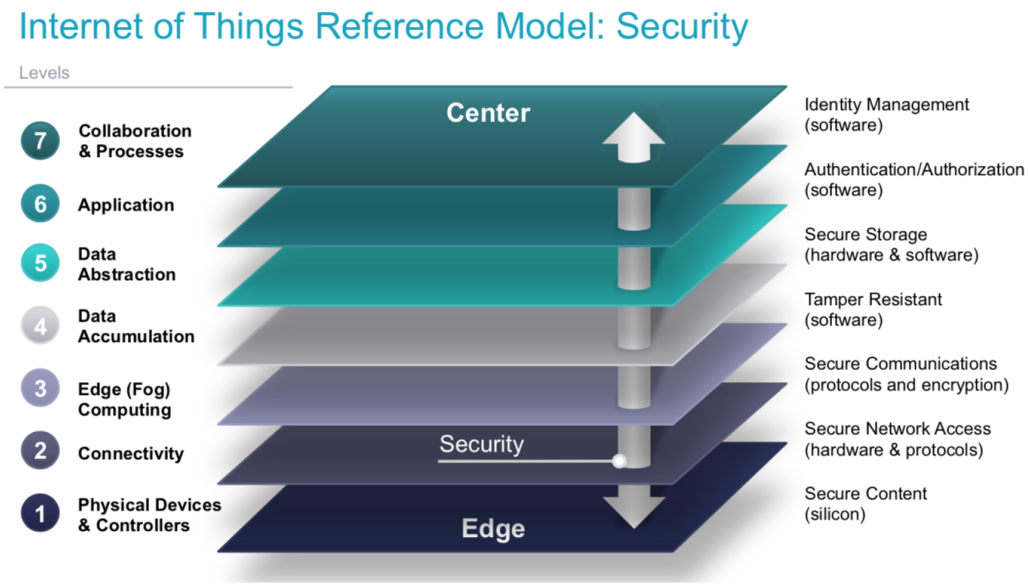

Figure 1: Cisco's View of the IoT Security Stack. This figure is from Cisco's Whitepaper [3].

3 Attacks on the IoT

Two well known attacks on IoT systems will be presented in this section: The DolphinAttack and the Mirai botnet.

3.1 DolphinAttack [4]

The DolphinAttack is an attack on physical IoT devices that issues inaudible, ultrasonic signals to voice controllable systems to control these devices in unwanted ways. With this capability, an attacker could engage in malicious behavior including executing unauthorized commands, modifying data, spying, and denial of service without the knowledge of the user of a device.

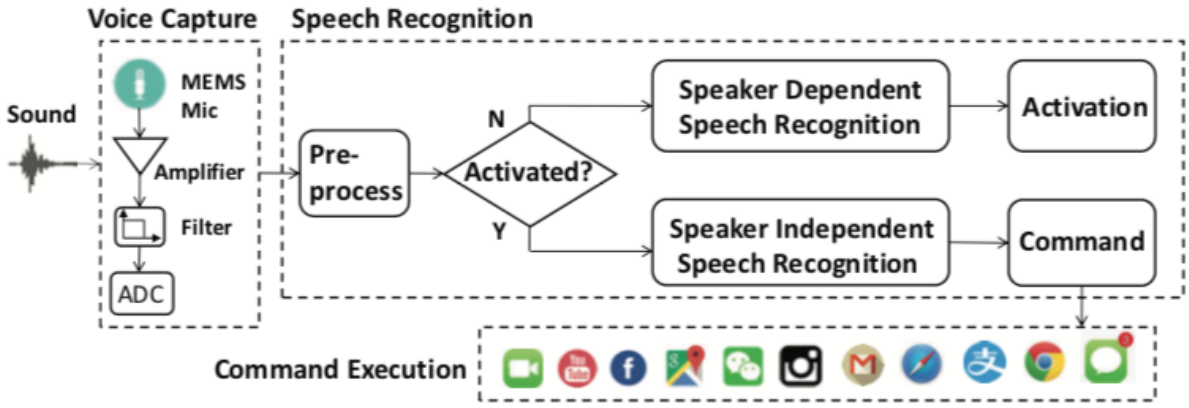

Figure 2: Diagram of a typical VCS system. This figure is from Zhang et al. [4].

3.1.1 Voice Controllable System

A voice controllable system (VCS) is a speech recognition system that responds to voice commands by a user. These systems are becoming increasingly popular because they can be operated hands free. Some examples of a voice controllable system are Apple Siri, Amazon Alexa, etc. As seen in Figure 2, a typical VCS has a few main modules: a voice capture module, a speech recognition module, and a command execution module. Also notable is that the speech recognition module usually needs to be activated with a speaker **dependent** speech recognition before it will process speaker **independent** commands. Most machine learning attacks target the speech recognition system, while malware attacks can maliciously mutate command execution logic. The DolphinAttack, however, focuses on exploiting limitations in the voice capture module of a VCS.

3.1.2 Threat Model

The assumption is that the attacker has no access to the target device and cannot interact with the user in any way. The attacker does know the technical specs of the target device, and the attacker has access to any equipment needed to construct the attack.

3.1.3 How the Attack Works

The attack works by modulating a base voice signal with an ultrasonic (high frequency) carrier signal. This results in a modulated signal that contains the command that was issued but one that humans cannot hear.

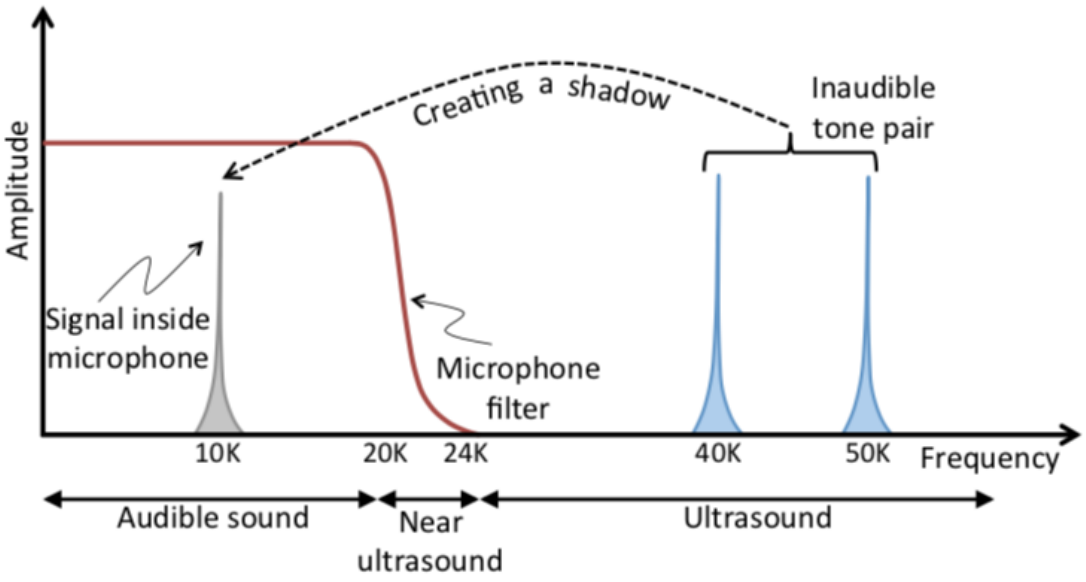

Figure 3: Behavior of voice capture module in creating a shadow signal. This figure is from Roy et al. [5].

Voice capture modules on many voice controllable systems are designed to screen out frequencies above the audible range to prevent any non-audible noise from being picked up. However, non linear behavior in components such as the amplifier (seen in Figure 2) actually create a shadow signal that is within the range of the filter even though the actual modulated signal is out of range [5]. This is demonstrated in Figure 3. These limitations in microphone design allow a voice controlled system to pick up commands from ultrasonic signals that are inaudible to the human ear.

A couple approaches can be used to trick the speaker dependent speech recognition system to activate the VCS. One approach uses a text to speech system to try to imitate a target user via brute force tries. This does not require any previous knowledge of the targeted user. Another approach concatenates voice recordings of the target user to craft a fake command. This requires previous recordings of the targeted user. Once activated, commands can be issued via speaker independent signals.

Zhang et al. demonstrate that the DolphinAttack can work on many real world devices, though its effectiveness varies drastically from device to device. For example, an Apple iPhone 4S can be activated from 110 centimeters away and can recognize commands up to 175 centimeters away, while an Apple iPhone 6 Plus can only be activated from 2 centimeters away and was not able to recognize any commands. The authors attribute this to variation in microphone and device design.

3.1.4 Defenses

Possible hardware defenses to the DolphinAttack include:

- enhanced microphones: Ideally, a microphone used in a VCS should not be able to sense sounds outside of the audible range.
- inaudible voice command cancellation: A component can be added in the voice capture module to detect AM modulated ultrasound frequencies and cancel out the baseband signal.

Possible software defenses to the DolphinAttack include:

- machine learning: Demodulated signals differ from original voice signals, and these differences could be detected with a machine learning classifier.

3.2 Mirai Botnet [6]

The Mirai botnet is an IoT attack that has been successfully launched in the real world numerous times and has affected a huge number of IoT devices. It is mainly a DDoS (Distributed Denial of Service) attack that uses subverted IoT devices to flood a target victim over the Internet.

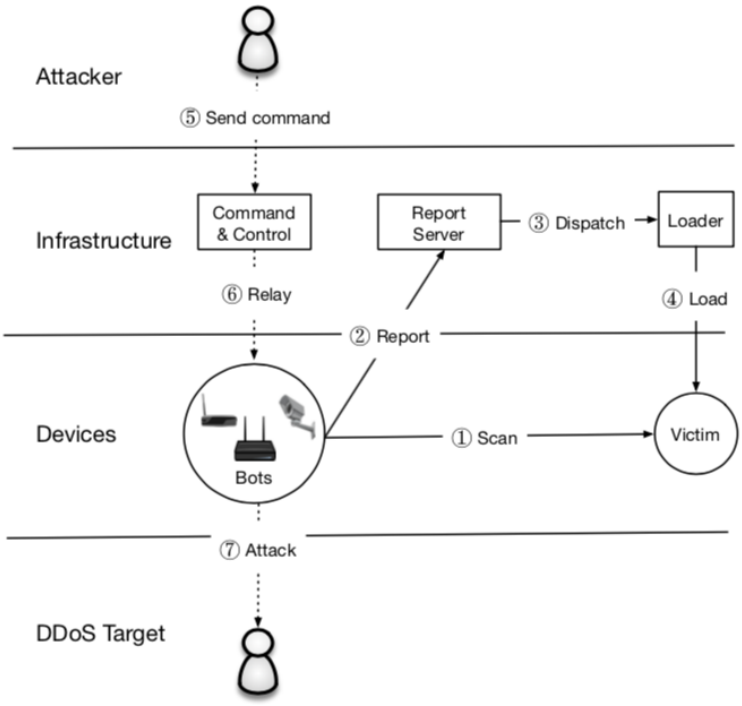

Figure 4: Overview of the Mirai botnet attack. This figure is from Antonakakis et al. [6].

3.2.1 Threat Model

The Mirai botnet assumes that an attacker has access to devices that can scan the Internet for IoT devices to infect. The attacker has familiarity with what ports and IP addresses that IoT devices may be listening on as well as knowledge about common IoT devices and their specifications. The attacker does not have physical access to the IoT devices it attacks because they could be located throughout the globe. The Mirai attack also assumes the attacker has enough compute resources to set up a system similar to the one in Figure 4 to infect devices and run the attack.

3.2.2 General Overview

The Mirai botnet attack takes advantage of the widespread use of insecure default passwords in IoT devices, a glaring security weakness. A general Mirai botnet attack can be broken down into seven steps that can be seen in Figure 4.

1. **Rapid Scanning Phase**
    - Using devices that are running the Mirai bot, the Mirai attack first asynchronously and statelessly sends TCP SYN packets to pseudorandom IPv4 addresses on Telnet TCP ports 23 and 2323. If the bot detects a victim, it enters step two.

2. **Brute-force Login Phase**

    - This phase is where a Mirai bot tries to establish a Telnet connection using a dictionary of default username and password pairs. On successful login, Mirai sends the victim IP and login credentials to a report server.

3. **Dispatch Phase**

    - A loader program is dispatched asynchronously by the report server to infect a victim device.

4. **Loader Program Phase**

    - The dispatched loader program infects a device with the Mirai bot by logging in, determining the system environment, and installing the appropriate malware. Once this happens, the victim device becomes a part of the Mirai botnet, repeating step 1 while also listening to a command and control server for instructions. Some forms of Mirai attempt to conceal the presence of Mirai malware by deleting downloaded binaries and randomizing its process name. This makes Mirai hard to track because the Mirai bot does not persist across reboots. Some more advanced forms of the Mirai bot even kill other processes bound to Mirai ports and kill other competing bots to make the attack more effective.

5. **Command Send Phase**

    - At some point, an attacker will decide they want to use all of the compromised IoT devices under their control to perform a DDoS attack on a target. They issue a command to a command and control server that will DDoS the target.

6. **Command Relay Phase**

    - The command and control server relays the attack command to all compromised devices.

7. **Attack Phase**

    - Compromised devices perform the specified attack.

3.2.3 Attack Instances

Of the many instances of the Mirai botnet attack that have been launched, three major occurrences of the attack in particular demonstrate its large power and scope.

1. **Krebs on Security**

    - Krebs on Security is a popular security blog that was subject to an unprecedented DDoS attack attributed to a Mirai botnet on September 21, 2016. The attack peaked out at 623 Gbps, with most of the IoT devices responsible for the attack were located in Southeast Asia and South America.

2. **Dyn**

    - Dyn, a popular DNS provider, was compromised by a Mirai botnet attack one month after the Krebs attack. This attack was comprised mostly of SYN floods on DNS and TCP ports and blocked name resolution for several popular sites such as Amazon, Github, and Netflix.

3. **Lonestar Cell**

    - Lonestar Cell is a large telecom operator in Liberia. This entity was subject to a Mirai attack that was strong enough to substantially deteriorate Internet connection in Liberia. This attack mostly comprised of SYN and ACK floods to Lonstar Cell IP addresses.

3.2.4 Mitigation

Some practices that could mitigate the effects of a Mirai botnet attack include:

- using basic password security practices like randomizing default passwords.
- choosing network configurations limiting remote access and adopting default closed-port practices instead of default open port.
- enforcing device security hardening practices such as ASLR (Address Space Layout Randomization), memory isolation boundaries, and principles of least privilege.
- implementing automatic updates; though this would require significant system design changes because most IoT devices do not support updating, it would help to fix compromised devices.
- out of band notifications that could notify users of the state of a device.
- implementing a way of identifying devices over the network such that their actions can be managed and track easier.
- defragmenting the IoT community such that there is more unified effort to combat Mirai attacks.
- adopting more secure practices for discontinuing support to old devices and device end of life.

It is important to note that mitigation of a Mirai botnet type attack is difficult because of the variety of device designs and device producers in IoT. Because IoT devices are heterogeneous by nature, there has not been much of a unified effort among the IoT community to focus on improving security. Another fact to note is that most Mirai botnet attacks have originated in Southeast Asia and South America with their target DDoS victims located in other parts of the world. Antonakakis et al. note that up to 50 percent of infected devices in the Krebs on Security attack mentioned in Section 3.2.3 were located in this region of Southeast Asia and South America. This geographic pattern associated with the Mirai attack shows that any solution to a Mirai botnet type attack needs to be global in nature. If any one section of the globe is lacking in IoT security, devices all around the world are prone to attack from Mirai.

References

[1] Apthorpe N., Reisman D., Sundaresan S., Narayanan A., Feamster N. *Spying on the Smart Home: Privacy Attacks and Defenses on Encrypted IoT Traffic*. 2017.

[2] F. Merces: Cryptocurrency-Mining Malware Targeting IoT, Being Offered in the Underground, `https://blog.trendmicro.com/trendlabs-security-intelligence/cryptocurrency-mining-malware-targeting-iot-being-offered-in-the-underground/`

[3] Cisco: The Internet of Things Reference Model, `http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf`

[4] Zhang G., Yan c., Ji X., Zhang T., Zhang T., Xu W. *DolphinAttack: Inaudible Voice Commands*. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (Dallas, TX, USA, 2017).

[5] Roy N., Hassanieh H., Choudhury R. R. *BackDoor: Making Microphones Hear Inaudible Sounds*. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services* (Niagara Falls, NY, USA, 2017).

[6] Antonakakis M., April T., Bailey, M., Bernhard M., Bursztein E., Cochran J., et al. *Understanding the Mirai Botnet*. In *Proceedings of the 26th USENIX Security Symposium* (Vancouver, BC, Canada, 2017).