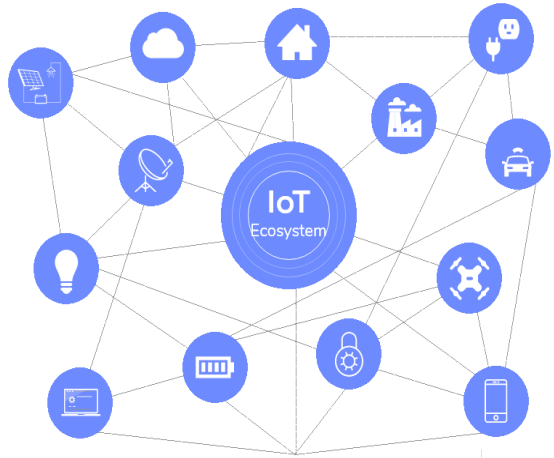
Unit 5:
IoT Forensics



Dr. Ujjaval Patel

Assistant Professor (IOT/SCADA)

ujjaval.patel@nfsu.ac.in

+91 987 987 9746

Unit Outlines:

- Introduction to IoT Forensics:
  - IoT Security
  - Security Problems
  - Attack Surface
- OWASP Vulnerabilities & its mitigation techniques
- Forensic Investigation of IoT Devices
- Forensic Tools & Techniques
- IoT Standards & guidelines
- Case Study

Acknowledgements:

- Open Web Application Security Project (OWASP)

Introduction to IoT Forensics : Security

What is Security?

- "The quality or state of being secure—to be free from danger"
- A successful organization should have multiple layers of security in place:
  - Physical security
  - Personal security
  - Operations security
  - Communications security
  - Network security
  - Information security

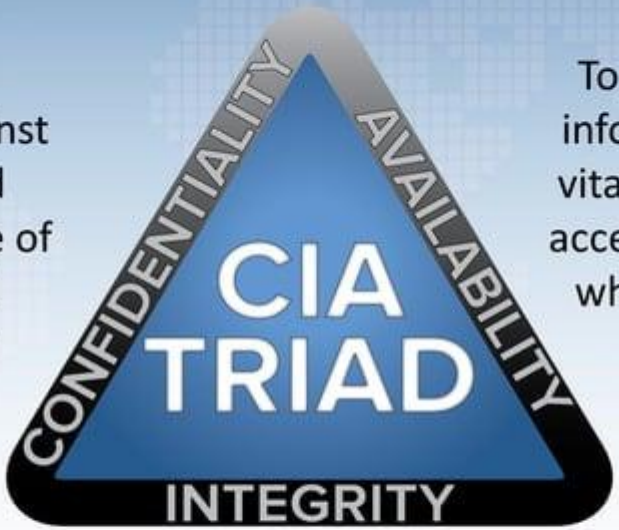
Introduction to IoT Forensics : Security

What is Information Security?

- The protection of information and its critical elements, including systems and hardware that use, store, and transmit that information
- Necessary tools: policy, awareness, training, education, technology

Confidentiality-Integrity-Availability (CIA)

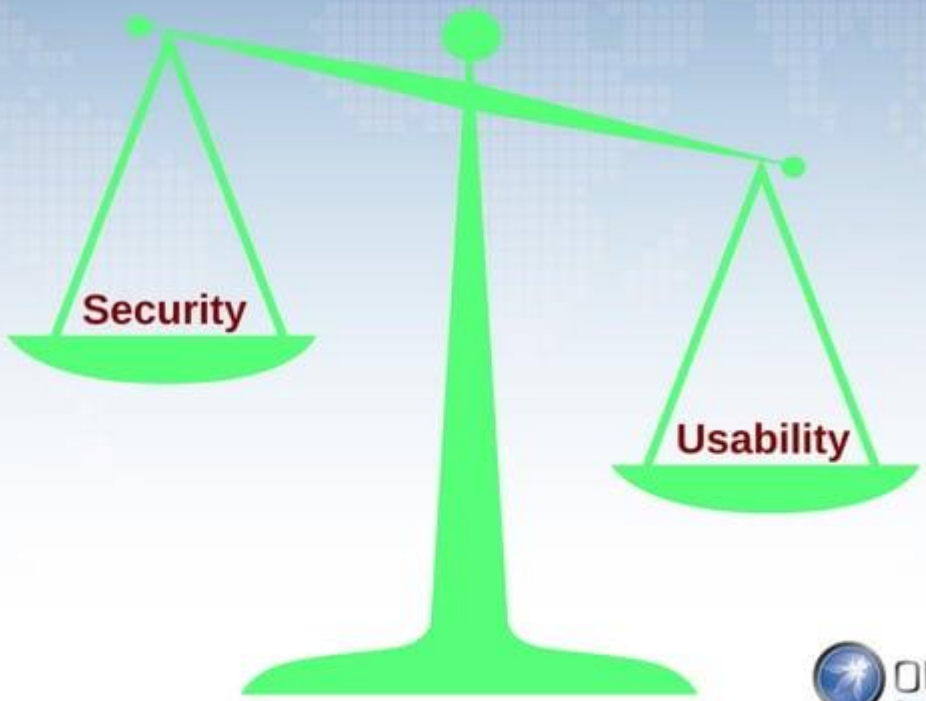
To ensure protection against unauthorized access to or use of confidential information



To ensure that information and vital services are accessible for use when required

To ensure the accuracy and completeness of information to protect business processes

Security vs. Usability



Security vs. Safety (General Usage)

- Security is concerned with malicious humans that actively search for and exploit weaknesses in a system.

Security vs. Safety (General Usage)

- Security is concerned with malicious humans that actively search for and exploit weaknesses in a system.
- Safety is protection against mishaps that are unintended (such as accidents)

Problems of IoT Security

- Initial design was for private communication network then moved to IP network and later on the Internet

Problems of IoT Security

- Initial design was for private communication network then moved to IP network and later on the Internet
- Firmware updates are hard or nearly impossible after installations

Problems of IoT Security

- Initial design was for private communication network then moved to IP network and later on the Internet
- Firmware updates are hard or nearly impossible after installations
- Started with basic security then found the security flaws and attached more complex security requirements later

Problems of IoT Security

- Initial design was for private communication network then moved to IP network and later on the Internet
- Firmware updates are hard or nearly impossible after installations
- Started with basic security then found the security flaws and attached more complex security requirements later
- Low security devices from early design are still out there and used in compatible fall-back mode

Flaw in Design

Home Hacking Tech Deals Cyber Attacks Malware Spying

The Hacker News™

Security in a serious way

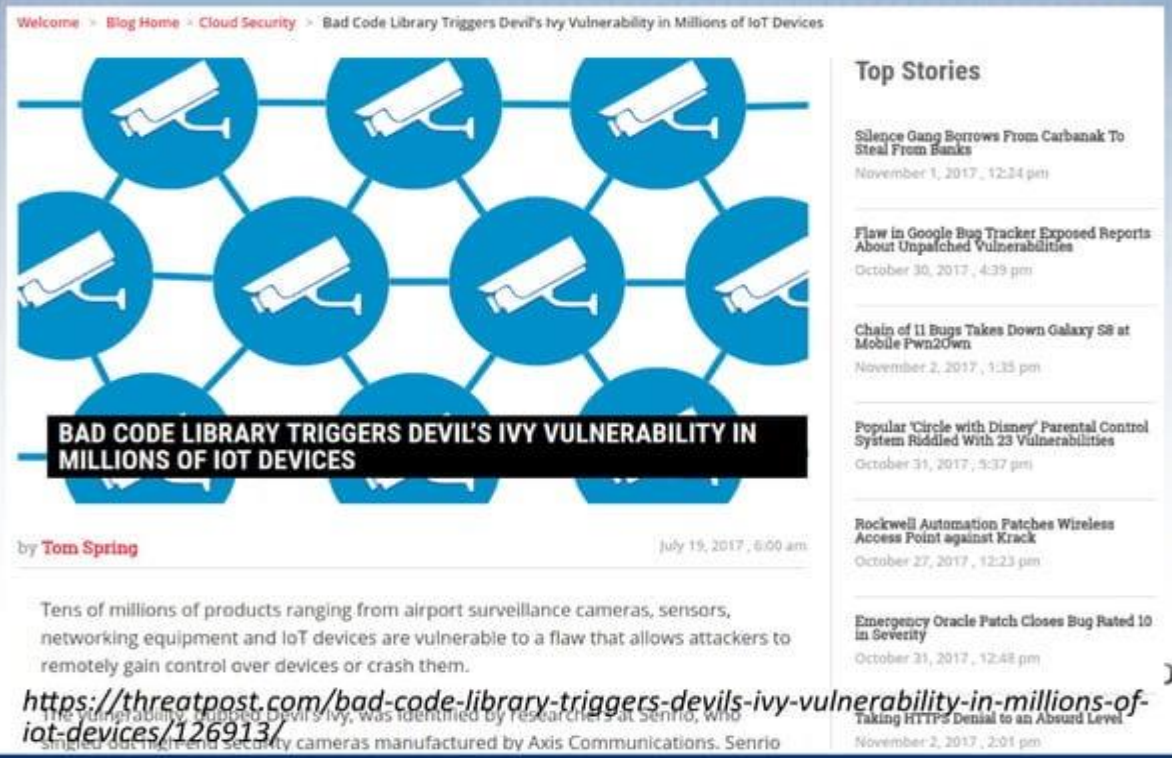
Unpatchable Flaw in Modern Cars Allows Hackers to Disable Safety Features

Thursday, August 17, 2017 Mohit Kumar

Tweet Share Share 48 Share 749 Share 1.34k Share

https://thehackernews.com/2017/08/car-safety-hacking.html

Flaw in Library

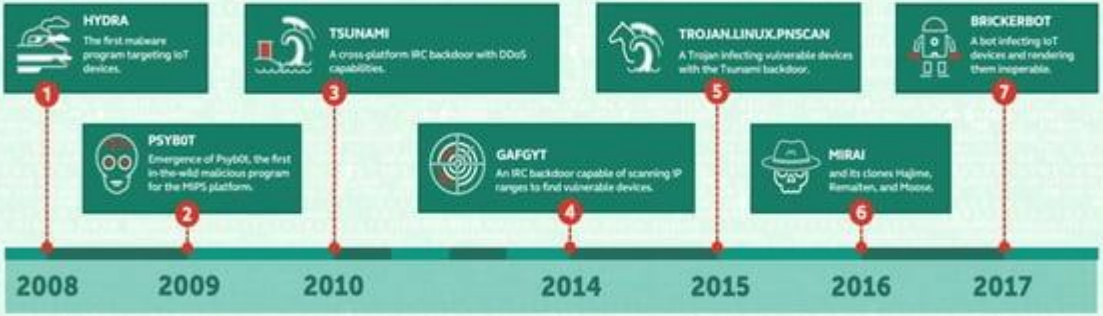


https://threatpost.com/bad-code-library-triggers-devils-ivy-vulnerability-in-millions-of-iot-devices/126913/

Rises of Threats Target IoT Devices

IoT devices at risk: malicious programs target the 'Internet of Things'

Currently, over 6 billion of 'smart' devices exist globally. It was when the Mirai botnet emerged in 2016 that the whole world learned how dangerous such devices may become in the hands of cybercriminals. However, the history of malware attacking IoT devices began much earlier.



KASPERSKY

https://securelist.com/honeypots-and-the-internet-of-things/78751/

OWASP Open Web Application Security Project

Types of IoT Classified by Communication

- Client Type
  - Most of implementation
  - e.g. payment terminal, IP Camera (call back to server), Smart Cars

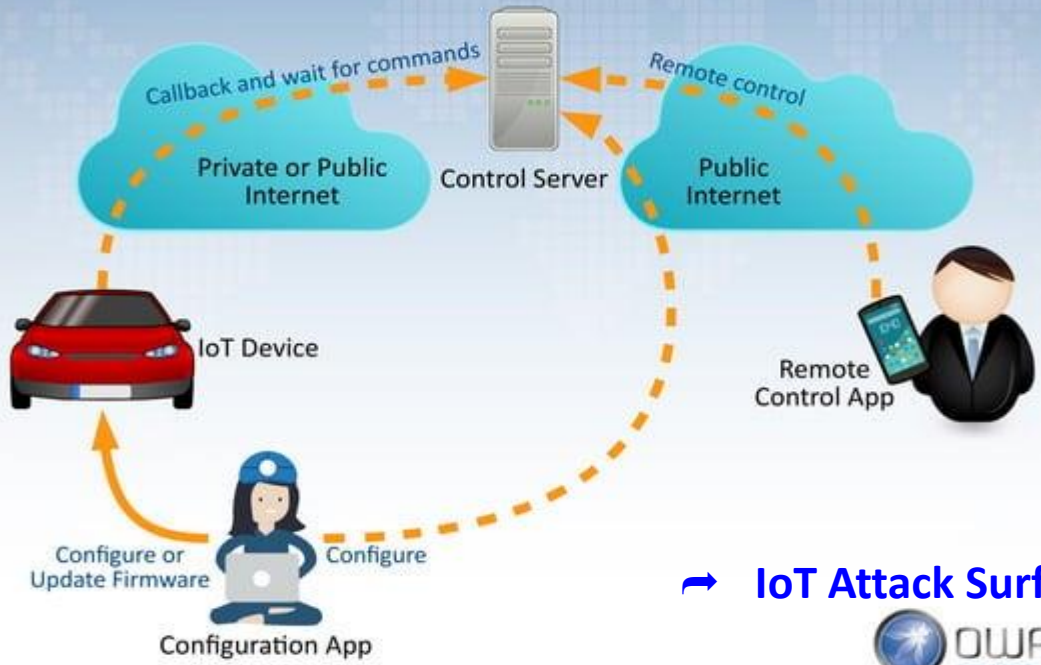
Types of IoT Classified by Communication

- Client Type
  - Most of implementation
  - e.g. payment terminal, IP Camera (call back to server), Smart Cars
- Server Type
  - e.g. IP Camera (built-in web interface)

Types of IoT Classified by Communication

- Client Type
  - Most of implementation
  - e.g. payment terminal, IP Camera (call back to server), Smart Cars
- Server Type
  - e.g. IP Camera (built-in web interface)
- Peer-to-Peer or Mesh

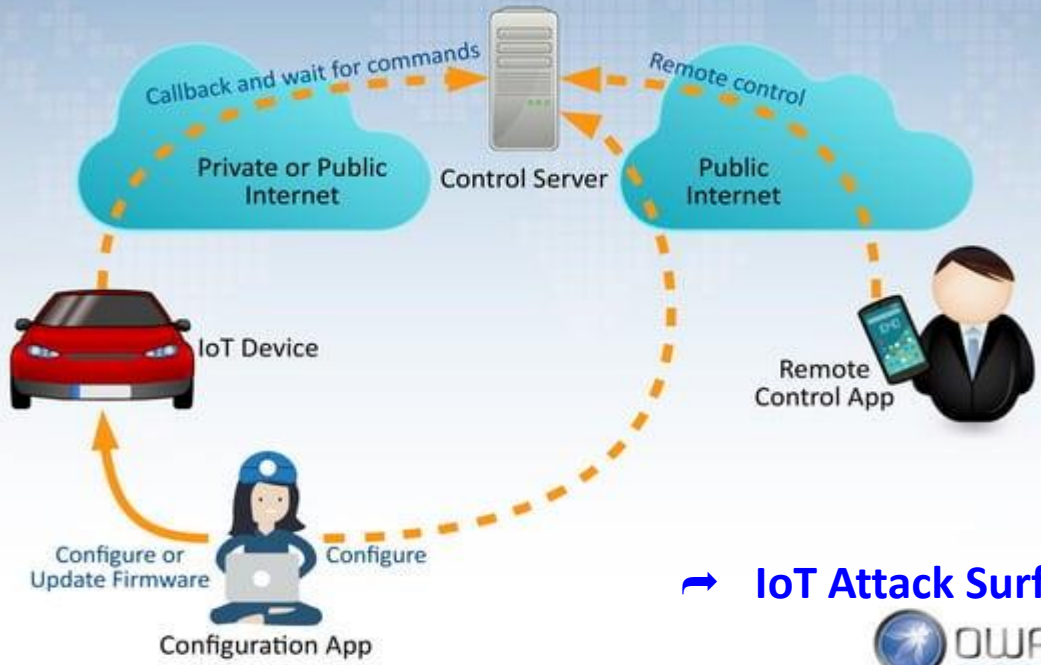
Typical IoT Infrastructure



➦ **IoT Attack Surface**

OWASP Open Web Application Security Project

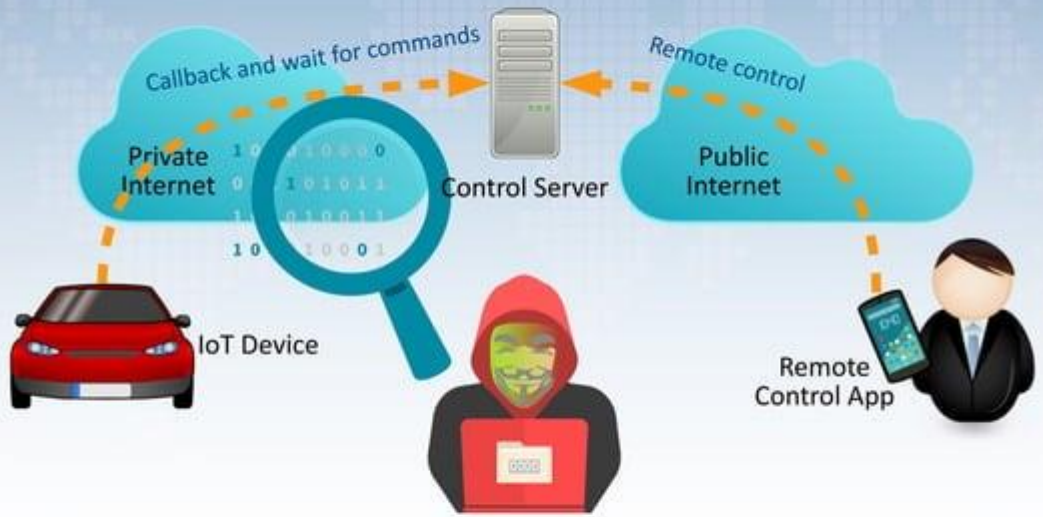
Typical IoT Infrastructure



➦ **IoT Attack Surface**

OWASP Open Web Application Security Project

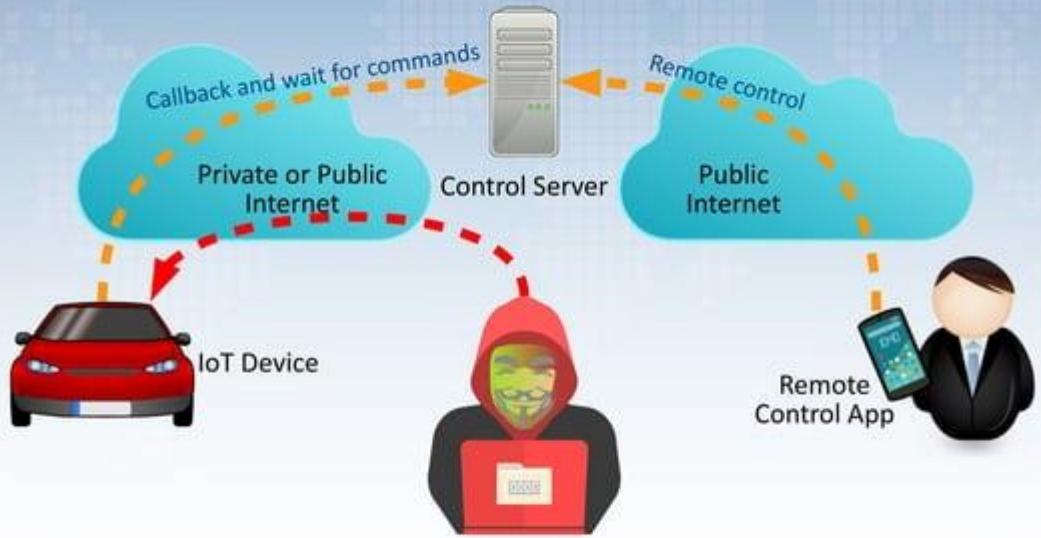
Typical Attack: Sniff Data on Private Network



Typical Attack: Fake Control Server

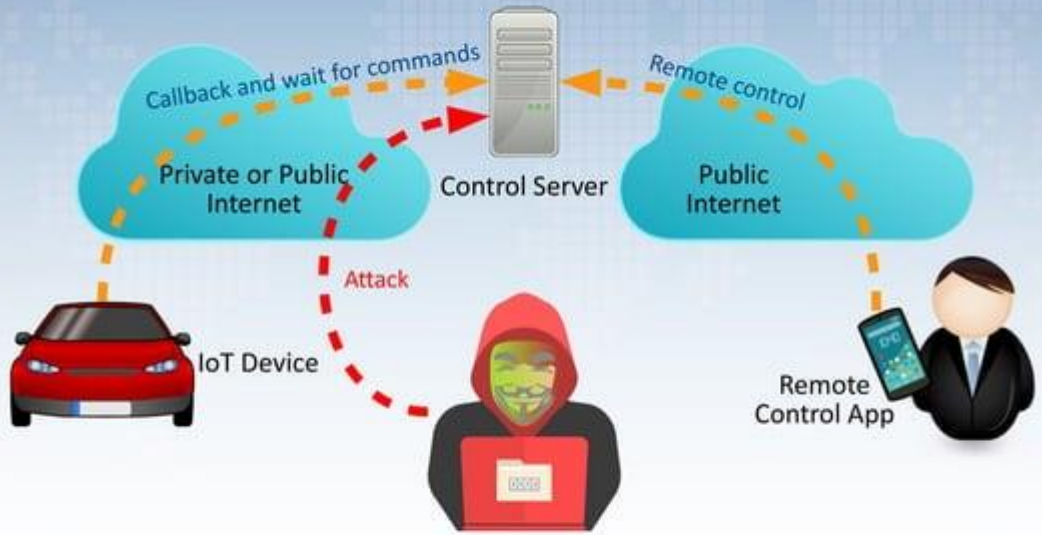
➦ IoT Attack Surface

Typical Attack: Attack on Device Open Ports



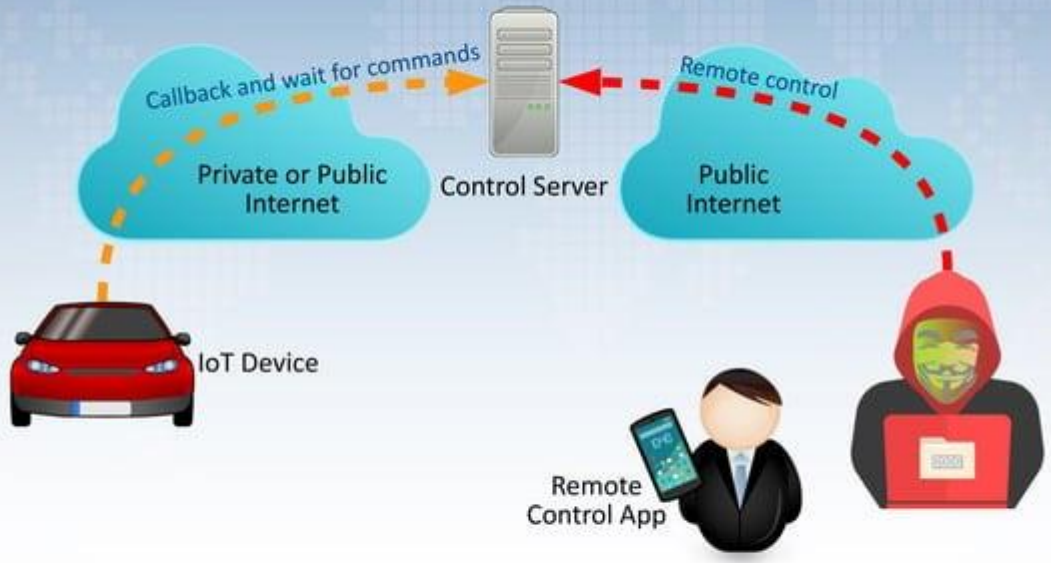
➦ **IoT Attack Surface**

Typical Attack: Attack on Server Open Ports



➦ **IoT Attack Surface**

Typical Attack: Steal Credential



➦ IoT Attack Surface

Typical Attack:
Inject Bad Configuration or Firmware

Other Attack Surface Areas → See OWASP

- Ecosystem
- Device Memory
- Device Physical Interfaces
- Device Web Interface
- Device Firmware
- Device Network Services
- Administrative Interface
- Local Data Storage
- Cloud Web Interface
- Third-party Backend APIs
- Update Mechanism
- Mobile Application
- Vendor Backend APIs
- Ecosystem Communication
- Network Traffic
- Authentication/Authorization
- Privacy
- Hardware (Sensors)

https://www.owasp.org/index.php/IoT_Attack_Surface_Areas

TOP 10

OWASP Top 10 IoT Vulnerabilities 2014

I1 Insecure Web Interface

I2 Insufficient Authentication/Authorization

I3 Insecure Network Services

I4 Lack of Transport Encryption/Integrity Verification

I5 Privacy Concerns

I6 Insecure Cloud Interface

I7 Insecure Mobile Interface

I8 Insufficient Security Configurability

I9 Insecure Software/Firmware

I10 Poor Physical Security

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

1 Insecure Web Interface

covers IoT device administrative interfaces

Obstacles

Default usernames and passwords

No account lockout

XSS, CSRF, SQLi vulnerabilites

Solutions

Allow default usernames and password to be changed

Enable account lockout

Conduct web application assessments

OWASP TOP 10 INTERNET OF THINGS

VULNERABILITY CATEGORIES

Insufficient Authentication/Authorization 2

covers all device interfaces and services

Obstacles

- Weak passwords
- Password recovery mechanisms are insecure
- No two-factor authentication available

Solutions

- Require strong, complex passwords
- Verify that password recovery mechanisms are secure
- Implement two-factor authentication where possible

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

3 Insecure Network Services

covers all network services including device, cloud, web and mobile

Obstacles	Solutions
Unnecessary ports are open	Minimize open network ports
Ports exposed to the internet via UPnP	Do not utilize UPnP
Network services vulnerable to denial of service	Review network services for vulnerabilities

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

Lack of Transport Encryption 4

covers all network services including device, cloud, web and mobile

Obstacles

Sensitive information is passed in clear text

SSL/TLS is not available or not properly configured

Proprietary encryption protocols are used

Solutions

Encrypt communication between system components

Maintain SSL/TLS implementations

Do not use proprietary encryption solutions

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

5 Privacy Concerns

covers all components of IoT solution

Obstacles

- Too much personal information is collected
- Collected information is not properly protected
- End user is not given a choice to allow collection of certain types of data

Solutions

- Minimize data collection
- Anonymize collected data
- Give end users the ability to decide what data is collected

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

Insecure Cloud Interface 6

covers cloud APIs or cloud-based web interfaces

Obstacles

- Interfaces are not reviewed for security vulnerabilities
- Weak passwords are present
- No two-factor authentication is present

Solutions

- Security assessments of all cloud interfaces
- Implement two-factor authentication
- Require strong, complex passwords

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

7 Insecure Mobile Interface

covers mobile application interfaces

Obstacles

- Weak passwords are present
- No two-factor authentication implemented
- No account lockout mechanism

Solutions

- Implement account lockout after failed login attempts
- Implement two-factor authentication
- Require stong, complex passwords

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

Insufficient Security Configurability 8

covers the IoT device

Obstacles

Password security options are not available

Encrytion options are not available

No option to enable security logging

Solutions

- Make security logging available
- Allow the selection of encryption options
- Notify end users in regards to security alerts

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

9 Insecure Software/Firmware

covers the IoT Device

Obstacles

- Update servers are not secured
- Device updates transmitted without encryption
- Device updates not signed

Solutions

- Sign updates
- Verify updates before install
- Secure update servers

OWASP TOP 10

INTERNET OF THINGS

VULNERABILITY CATEGORIES

Poor Physical Security 10

covers the IoT device

Obstacles

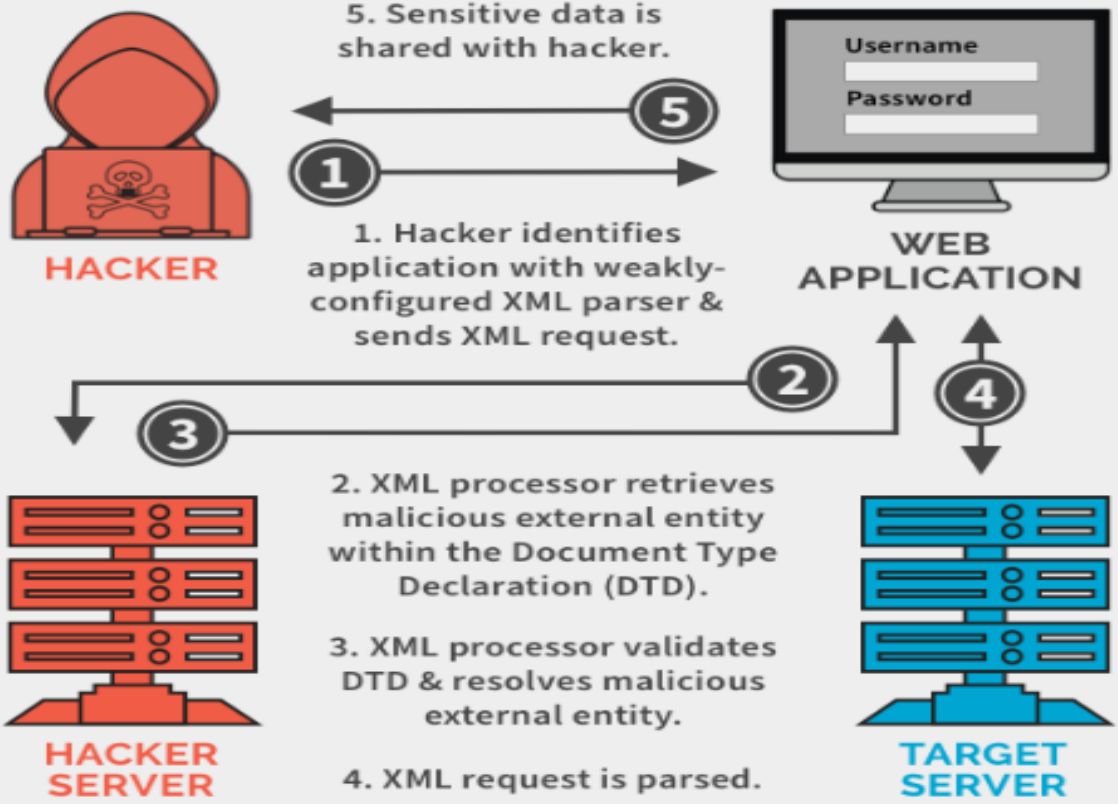
- Unnecessary external ports like USB ports
- Access to operating systems through remove media
- Inability to limit administrative capabilities

Solutions

- Minimize external ports like USB ports
- Properly protect operating system
- Include ability to limit administrative capabilities

Vulnerability/Risk	Description
1. Weak, Guessable, Hardcoded Passwords	Using easily brute-forced, publicly available, or unchangeable credentials
2. Insecure Network Services	Unneeded or insecure network services running on the device itself, especially those exposed to the internet, compromise the C.I.A. of information or allow unauthorized remote control
3. Insecure Ecosystem Interfaces	Insecure web, backend API, cloud, or mobile interfaces in the ecosystem outside of the device that allows compromise of the device or its related components.
4. Lack of Secure Update Mechanism	Lack of ability to securely update the device. Examples include lack of firmware validation on device, lack of secure delivery (plaintext transmission), lack of anti-rollback mechanisms
5. Use of Insecure or Outdated Components	Using deprecated or insecure software components/libraries that could allow the device to be compromised. Includes insecure customization of OS platforms, using third-party software, etc.
6. Insufficient Privacy Protection	User's personal information is stored on the device and is used insecurely or without permission
7. Insecure Data Transfer and Storage	Lack of encryption or access control of sensitive data anywhere within the ecosystem, including at rest, in transit, or during processing
8. Lack of Device Management	Lack of security support on devices deployed within production, including asset management, update management, secure decommissioning, systems monitoring, and response capabilities
9. Insecure Default Settings	Devices or systems shipped with insecure default settings or lack the ability to make the system more secure by restricting operators from modifying configurations
10. Lack of Physical Hardening	Lack of physical hardening measures, allowing potential attackers to gain sensitive information that can help in future remote attacks or take local control of the device

XML External Entity Attack (XXE)



Preventing an XXE Attack

A few pointers to secure your web apps from XXE attacks:

- **Parse the Parser**: Essentially, XXE is a form of injection attack that attacks weak XML parsers. Hence, a basic defense is to check your application's XML parsing library for XML features that can be misused, and disable them. In particular, disable DTDs (External Entities), as detailed here. Also, check that your XML processors are patched.
- **Verify Inputs:** Verify that file uploads are XSD validated, and only whitelisted URLs are allowed.
- **Test via Code**: Don't underestimate manual code reviews. Identify and test for XXE attacks via API calls. Validate user inputs prior to it being parsed by the XML parser.
- **Remember the Basics**: Ensure that network monitoring tools and application firewalls are updated.

https://owasp.org/www-project-top-ten/2017/A4_2017-XML_External_Entities_(XXE)

Mirai Malware

- Malware that turns networked devices running Linux into remotely controlled "bots" that can be used as part of a botnet in large-scale network attacks
- Primarily targets online consumer devices such as IP cameras and home routers using a table of more than 60 common factory default usernames and passwords, and logs into them to infect them with the Mirai malware.
- First found in August 2016
- Use in DDoS attacks
  - 20 September 2016 on the Krebs on Security site which reached 620 Gbit/s and 1 Tbit/s attack on French web host OVH
  - 21 October 2016 multiple major DDoS attacks in DNS services of DNS service provider Dyn
  - November 2016 attacks on Liberia's Internet infrastructure
- The source code for Mirai has been published in hacker forums as open-source

What Can We Learn from Mirai Attacks?

- Do not use default passwords for all default usernames
- If possible, do not allow configuration interface from Internet side
- If the IoT devices are used only in the organization, do not expose to the public Internet
- If there is a need to use from the Internet, open only necessary ports and use non-default ports where possible

Case: Dyn Botnet DDoS Attack

- DDoS Attack in October, 2016 → Target: DNS provider **Dyn**
  - DDoS attack was staged and launched from IoT devices using the Mirai malware
- **Mirai was designed for two main purposes:**
  - Find and infect IoT devices to grow the botnet
  - Participate in DDoS attacks based on commands received by remote Command and Control (C&C) infrastructure
- **Mirai operates in three stages:**
  1. Infect the device
  2. Protect itself
  3. Launch attack

Case: Dyn Botnet DDoS Attack (Cont.)

Stage 1:

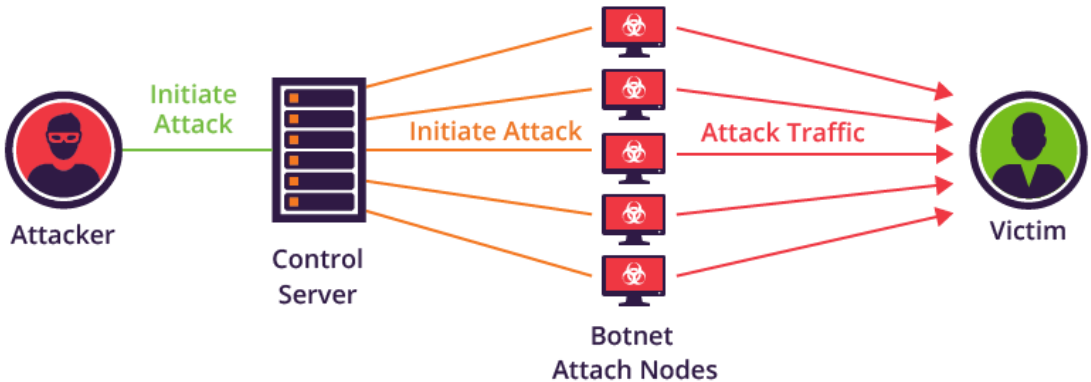
- Scan for IoT devices that are accessible over the Internet
  - Primarily scans for ports **22, 23, 5747,** etc. that are open
  - Can be configured to scan for others
- Once connected → brute-forces usernames and passwords to login to the device
- Use the device to scan networks looking for more IoT devices

Case: Dyn Botnet DDoS Attack (Cont.)

- **Stage 2: Protect itself**
  - Kill other process running on infected device (SSH, Telnet, HTTP) to prevent owner from gaining remote access to device while infected
  - Note: Rebooting the device can remove the malware, but it can become infected again

- **Stage 3: Launch attack**
  - Infected device launches different types of attacks
  - HTTP floods, SYN floods, etc. → DDoS-based attacks

Case: Dyn Botnet DDoS Attack (Cont.)



https://www.imperva.com/blog/how-to-identify-a-mirai-style-ddos-attack/

IOT Forensics

- The IoT Forensics could be perceived as a subdivision of the Digital Forensics.
- IoT Forensics is a relatively new and unexplored area.
- The purpose of the IoT Forensics is similar to the one of the Digital Forensics, which is to identify and extract digital information in a legal and forensically sound manner.

IoT Device Penetration Testing

PENETRATION TESTING

Penetration testing (also known as a “pen test”), is an authorized simulated attack on a computer system, designed to evaluate the security of the system. The test is performed to identify weaknesses including the potential for unauthorized parties to gain access to the system's features and data, as well as strengths, enabling a full risk assessment to be completed.

TYPES OF PENETRATION TESTS

NETWORK PENETRATION TEST

- BLACK BOX
- WHITE BOX
- GRAY BOX

WIRELESS PENETRATION TEST

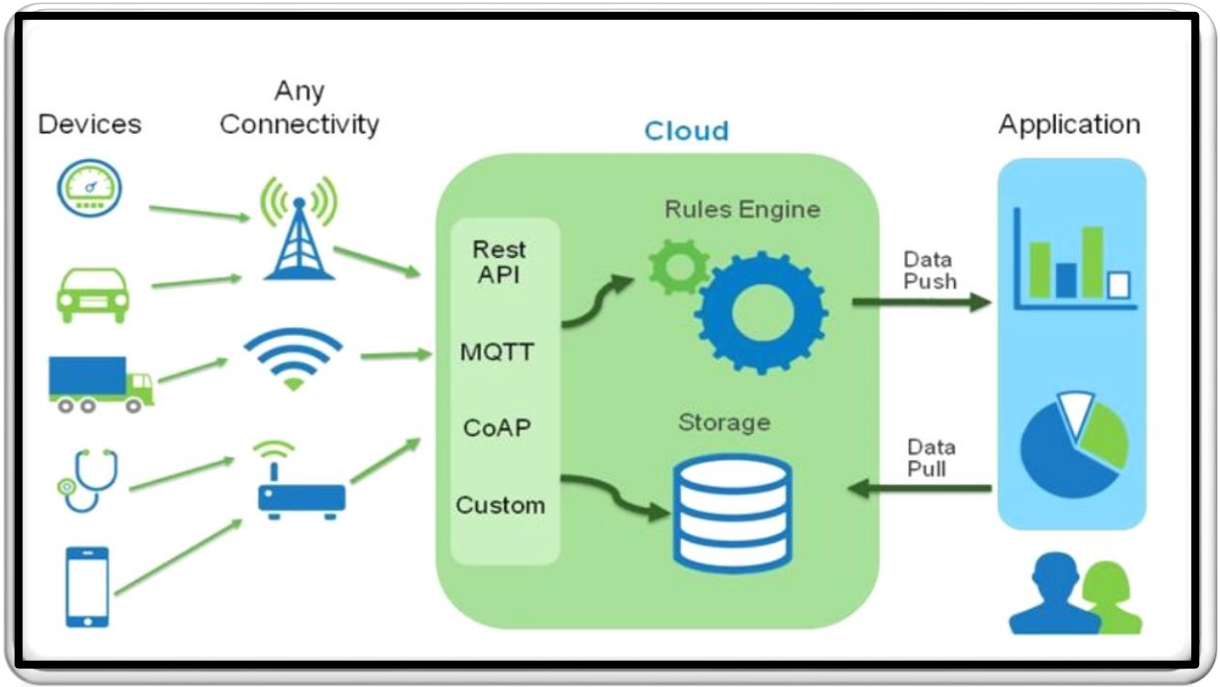
APPLICATION SECURITY TESTING

PHYSICAL PENETRATION TEST

SOCIAL ENGINEERING

- REMOTE
- PHYSICAL

How IoT Works



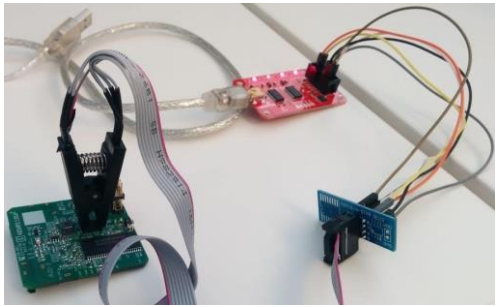
The Attack Vectors

- **Hardware**
- **Firmware**
- **Network**
- **Wireless Communications**
- **Mobile and Web applications**
- **Cloud API's**

IoT Device hardware pentest

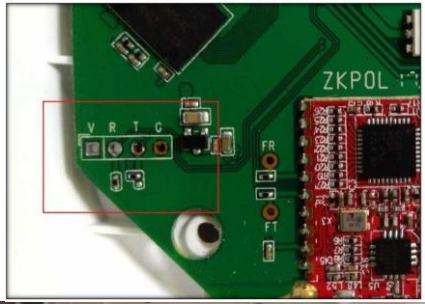
- Internal communications Protocols like UART,I2C, SPI etc.
- Open ports
- JTAG debugging
- Exacting Firmware from EEPROM or FLASH memory
- Tampering

IoT Pentesting Methodologies

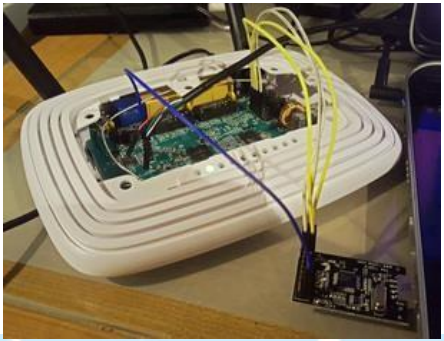


Source :
FireEye

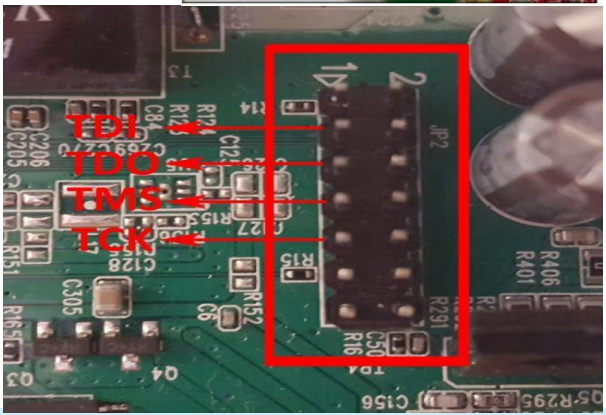
Dumping flash Memory



Open UART ports



JTAG Exploitation



Firmware Penetration testing

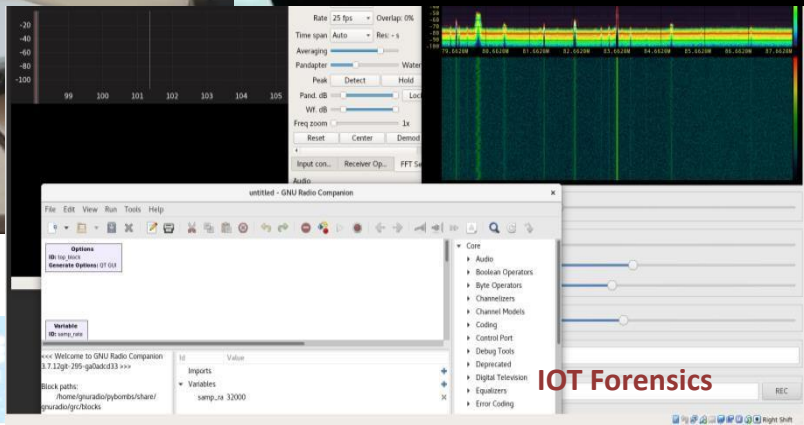
- Binary Analysis
- Reverse Engineering
- Analyzing different file system
- Sensitive key and certificates
- Firmware Modification

Radio Security Analysis

- Exploitation of communication protocols
  - ✓ BLE,Zigbee,LoRA,6LoWPAN
- Sniffing Radio packets
- Jamming based attacks
- Modifying and replaying packets

Analysis of radio signals using USRP

Universal Software Radio Peripheral (USRP)



Mobile, Web and Cloud Application Testing

- **Web dashboards: XSS, IDOR, SQL Injections**

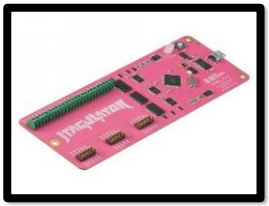
(Cross site scripting (XSS), **Insecure direct object references** (IDOR))

- **.apk and .ios Source code review**
- **Application reversing**
- **Hardcoded API keys**
- **Cloud Credentials like MQTT, CoAP, AWS etc.**

Software Tools

Hardware Level	Firmware Level	Radio Security
Baudrate.py	Binwalk	Gatttool
Esptool	Strings	hcitool
Flashrom	IDAPro	GNURadio
Minicom	Radare2	Killerbee
Screen	Qumu	

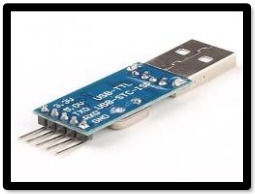
Hardware Tools



Jtagulator

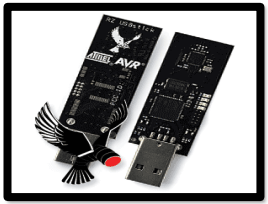
HackRF

Ubertooth

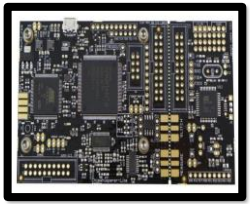


TTL-USB Converter

Bus Pirate



Zigbee Sniffer



Chip whisperer

Smart Lock Disclosure

FB50 Smart Lock Vulnerability Disclosure (CVE-2019-13143)

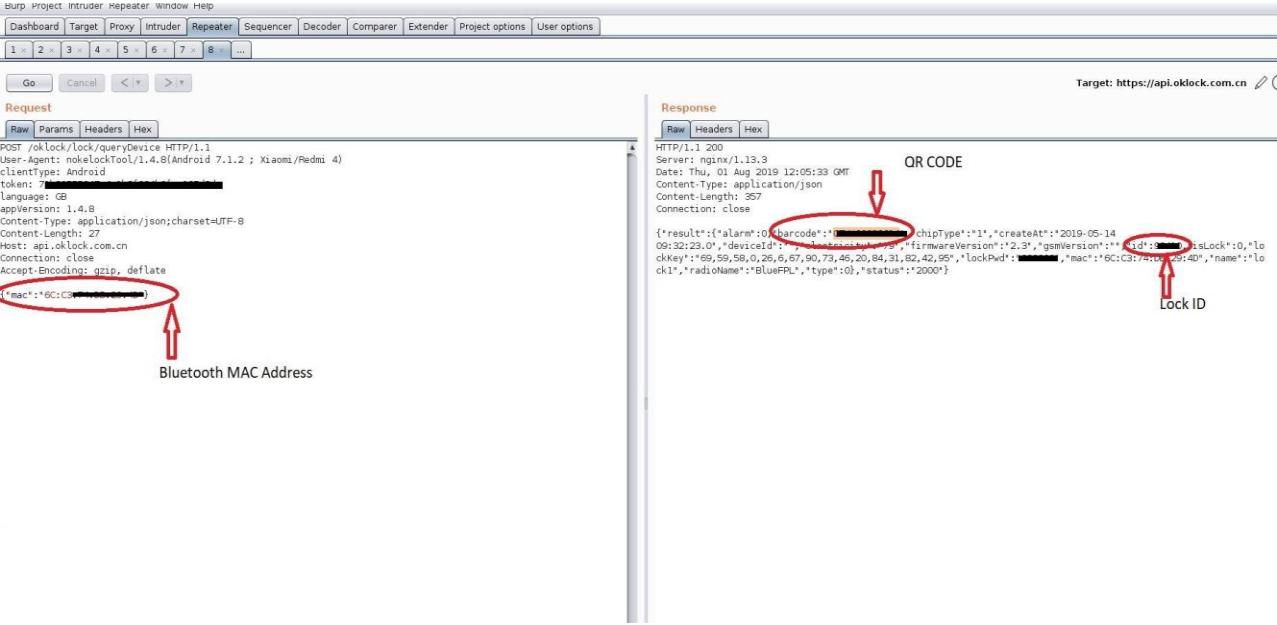
Posted on August 2, 2019 by Shubham Chougule

Executive Summary

Our security engineers found vulnerabilities in the FB50 smart lock mobile application. An information disclosure vulnerability chained together with poor token management lead to a complete transfer of ownership of the lock from the user to the attacker's account.

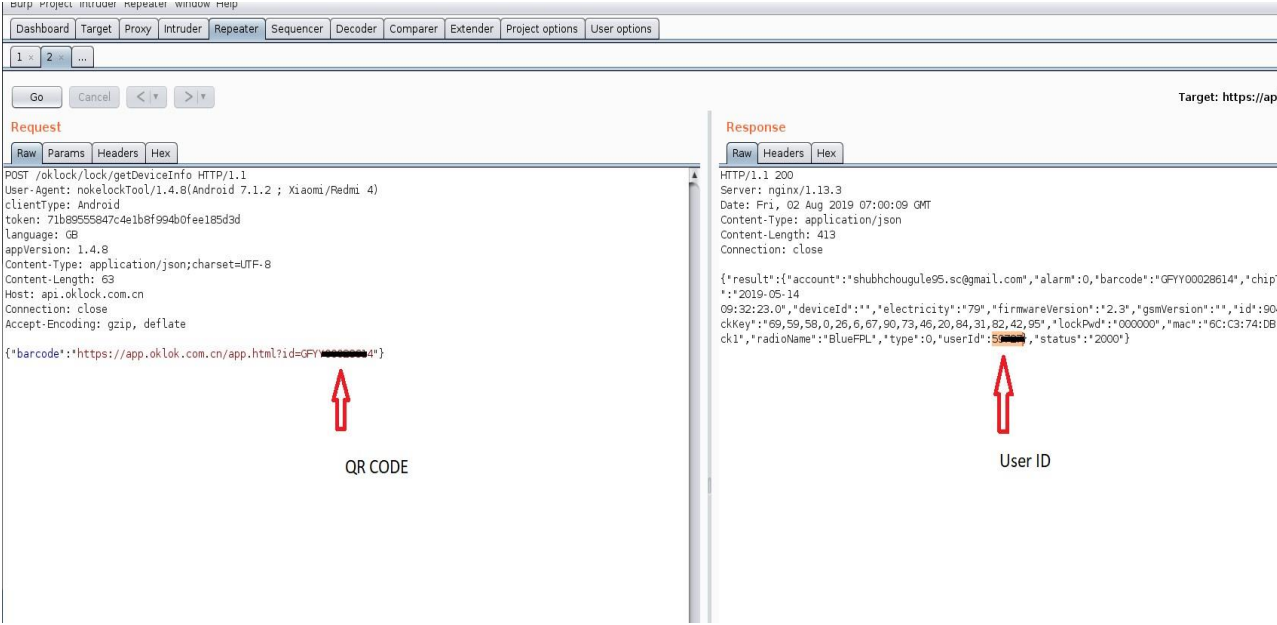
Smart Lock Disclosure

Getting QR code and Lock ID



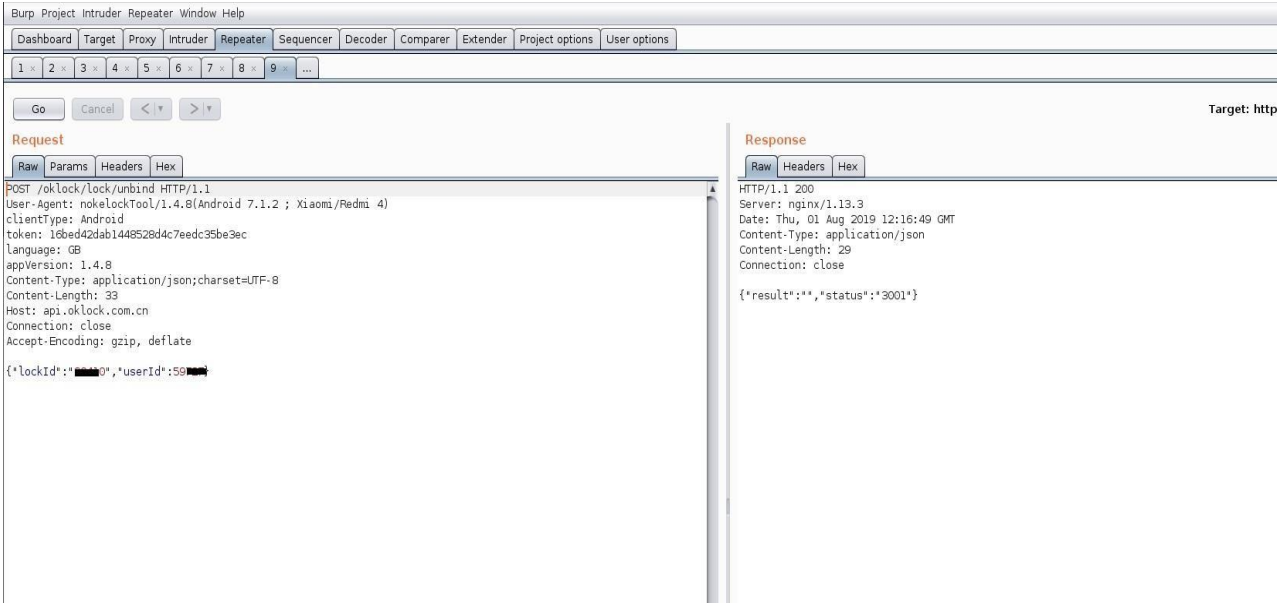
Smart Lock Disclosure

Getting the USER ID



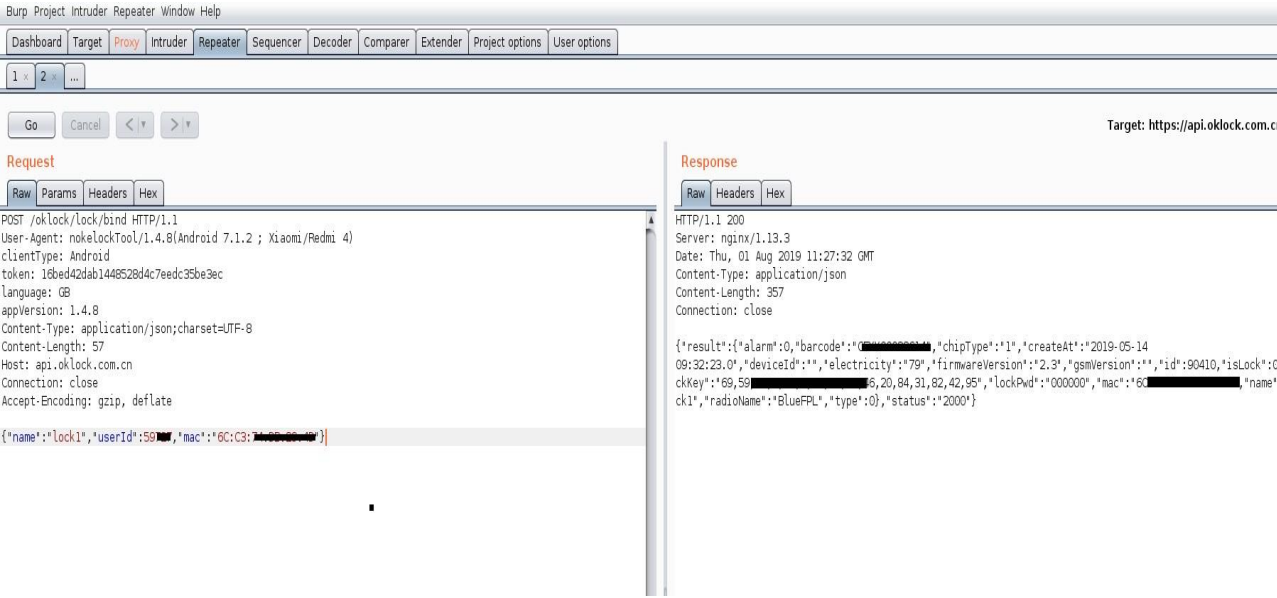
Smart Lock Disclosure

Unbind the Lock from victim's account



Smart Lock Disclosure

Bind the Lock to attacker's account



Best Practices

- Make hardware tamper resistant
- Provide for firmware updates/patches
- Specify procedures to protect data on device disposal
- Use strong authentication
- Use strong encryption and secure protocols
- Specify Destroy method if device get break down.

The need of IOT forensics

- ***Extensive attack surface***

  Despite all the benefits and the wide prospects of IOT, some IOT technologies are particularly vulnerable to cyber-attacks.

  IoT devices with public interfaces are exposed to greater risk levels because they could bring a malware to the private network from a less secure public space [2].

- ***New cyber-physical security threats***

  Using IoT technology, virtual crimes could step across the limit of cyberspace and threaten human life.

  E.g.: US FDA published a warning that certain pacemaker models are vulnerable to hacking .

- ***Contains Digital traces***

  IoT devices which is able prove or disprove certain hypothesis, and could, help the forensics professionals find answers and reconstruct the crime scene [3].

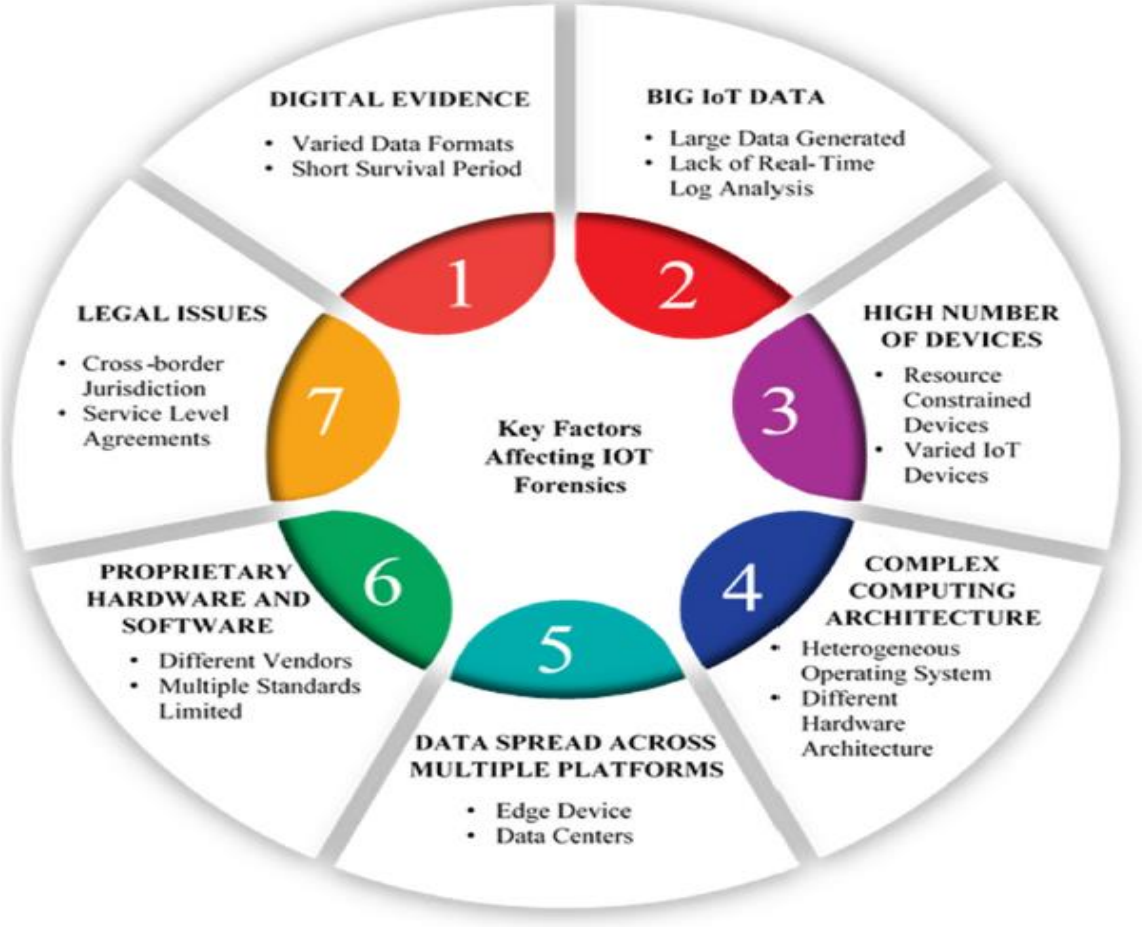
Categories of Evidences With Respect To A Crime Scene

1. **Smart devices and sensors :** It includes sensors, smart devices, automation tools those are powered by IoT Architecture, in other words, **the gadgets those are present in the Crime Scene**.

2. **Hardware and Software : Communication link** between smart devices and the external world which includes IPS, Firewalls, Computers.

3. **External resources : Areas outside networks under investigation**, that includes Cloud, Social Media, ISPs, Network Providers.

Reference:- https://hub.packtpub.com/iot-forensics-security-connected-world/


DIGITAL EVIDENCE
• Varied Data Formats
• Short Survival Period
1
BIG IoT DATA
• Large Data Generated
• Lack of Real-Time Log Analysis
2
HIGH NUMBER OF DEVICES
• Resource Constrained Devices
• Varied IoT Devices
3
COMPLEX COMPUTING ARCHITECTURE
• Heterogeneous Operating System
• Different Hardware Architecture
4
DATA SPREAD ACROSS MULTIPLE PLATFORMS
• Edge Device
• Data Centers
5
PROPRIETARY HARDWARE AND SOFTWARE
• Different Vendors
• Multiple Standards Limited
6
LEGAL ISSUES
• Cross-border Jurisdiction
• Service Level Agreements
7
Key Factors Affecting IOT Forensics

Challenges in IoT Forensics

- The IoT Forensics field is encountering an array of challenges, none of which has a simple solution.

  - General Issues.
  - Evidence Identification, Collection and Preservation issues.
  - Evidence Analysis and Correlation.
  - Presentation

General Issues

- Lack of a methodology and framework for IoT forensics.
- There is a lack of appropriate tools for IoT forensics.

Impact

- Could contaminate or destroy evidence.
- Absence of common forensic model could jeopardize the trust and agreements in cross jurisdictional investigations.

Evidence Identification, Collection and Preservation issues

- Detecting the presence of IoT systems, and identification of IoT devices that can provide evidence in an investigation.
- Lack of training for first responders.
- Wide range of software and/or hardware specifications.
- Lifespan limitations

Evidence Identification, Collection and Preservation issues

Impact

- Data could be easily overwritten.
- The responding officers often neglect or shut down the system directly, without first creating the necessary forensic image.

Evidence Analysis and Correlation

- Overwhelming amount of data that an IoT system might produce.
- Time Lining and Limited Correlation of Evidence.
- Data provenance -Less certainty about data ownership and modification history.
- Metadata – vast majority of IOT devices do not store any metadata.

Evidence Analysis and Correlation

Impact

- The amount can be overwhelming for an investigator and the tools used.
- Creating a time-line can be challenging.

Presentation

- Jury most probably has only basic understanding of cloud computing and forensics.
- It would be a challenging task to explain to them the technicalities behind such a complex architecture in the very limited time of the trial.
- Will the court accept the methodology and tools used since they are not yet standardized.

Presentation

Impact

- if an investigative body chooses unsuitable methods for acquisition it can harm the data integrity and can easily be challenged in Court due to omissions in the way of collection.

IOT FORENSICS METHODS AND TOOLS

- There are very few tools designed specifically for IOT forensics
- There is no unique methodology to investigate in a IOT environment
- None of the approaches has been widely accepted by the forensics community.
- Most of the approaches are still of theoretical nature.

Perform standard data acquisition

- Various proven techniques and procedures for Digital forensics are still applicable to IoT devices.
- if an IoT device can be connected to a computer, the internal storage of the device can be forensically imaged.
- Various Digital forensic tools are available to perform standard data acquisitions

  e.g.:
  - FTK Imager
  - X-ways forensics
  - ENCASE
- More practical and cost effective method [4], [5]

Perform standard data acquisition

Limitations

- In IoT forensics, where traditional investigative techniques & tools have a very low success rate.
- Useless against Proprietary echo systems e.g.: apple, amazon.
- Occasionally, formats of the collected data are invalid or vendor specific.

Interface testing

- Most IoT devices have web interfaces.
- Can get general knowledge of how the system works
- By testing the interface investigators can validate whether the relevant digital evidence is present and its condition.
- Investigators can identify any indicators of compromise.

Limitations

- Can lead to accidental contamination of evidence.

Oxygen Forensic Detective

- Oxygen Forensic Detective is an all-in-one forensic software platform.
- Able to extract, decode, and analyze data from multiple digital sources mobile and IoT devices, device backups, drones, and cloud service.
- Oxygen Forensic offers data extraction from two popular IoT devices based on Amazon & Google.
- Can performs logical acquisition from smart-wearables (apple watch, Fitbit, Samsung Health).

Oxygen Forensic Detective

Limitations

- Its support for range of devices is limited.
- It uses a brute force technique which can take a lot of time to complete the process.
- Oxygen Forensic suite is very expensive.

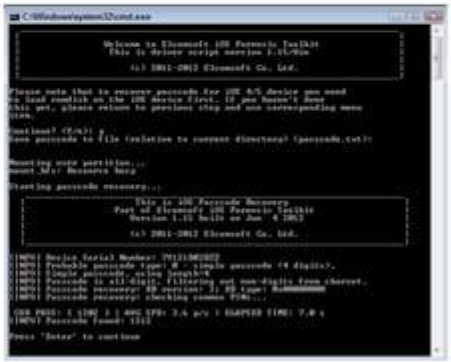
OXYGEN FORENSICS

Elcomsoft iOS Forensic Toolkit

- Perform full file system and logical acquisition for Apple ecosystem devices.
- The toolkit provides jailbreak-free forensic extraction
- Inbuilt write blocker.

Limitations

- Only supports Apple devices



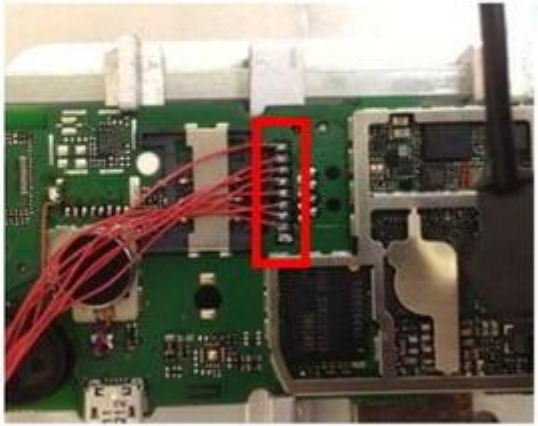
Firmware data extraction by JTAG

- JTAG stands for Joint Test Action Group is a common hardware interface that provides a way to communicate directly with the chips on a board.
- The port was initially designed for testing PCB (Printed Circuit Boards).
- JTAG Forensics involves acquiring firmware data using standard Test Access Ports (TAPs).
- Doesn't require specific data cables for each make/model.
- The data is transferred in a raw format.
- Able to recover data from damaged devices. [5]

Firmware data extraction by JTAG

Limitations

- it's difficult to find out the entire JTAG pin
- Not all Devices have a JTAG enabled chip.
- Forensics image creating process is slow.
- Need expert knowledge in electronics.



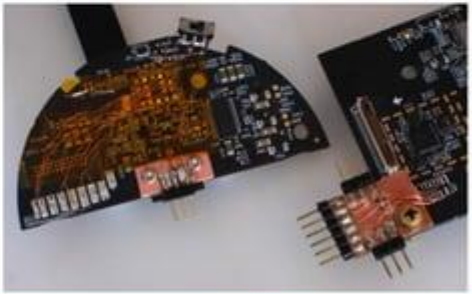
Firmware data extraction by UART

- UART is Universal Asynchronous Receiver/Transmitter.
- UART is a widely used method.
- It is a hardware device which is a part of Integrated circuitry and used for serial communications.
- UART converter translates serial data into readable data via USB.
- Hardware complexity is low. [5]

Firmware data extraction by UART

Limitations

- It can accidently reset the devices to factory settings resulting in loss of data.
- Size of a data frame is limited to 9 bits.



Cloud-based IoT Forensic Toolkit

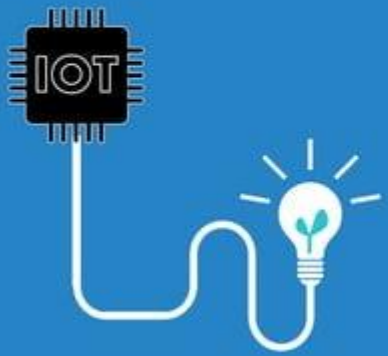
- Alexa is a cloud based assistant, it manages through a mobile application or the web.
- Previously, methods such as disassembling the Amazon Echo device and unofficial Alexa APIs to access cloud data were used.
- Researchers utilized mobile applications and web browsers to retrieve additional artifacts from the client to automate this process of data collection, visualization and evaluation [8]

Cloud-based IoT Forensic Toolkit

Limitations

- New technology.
- Concerns about evidence Integrity.

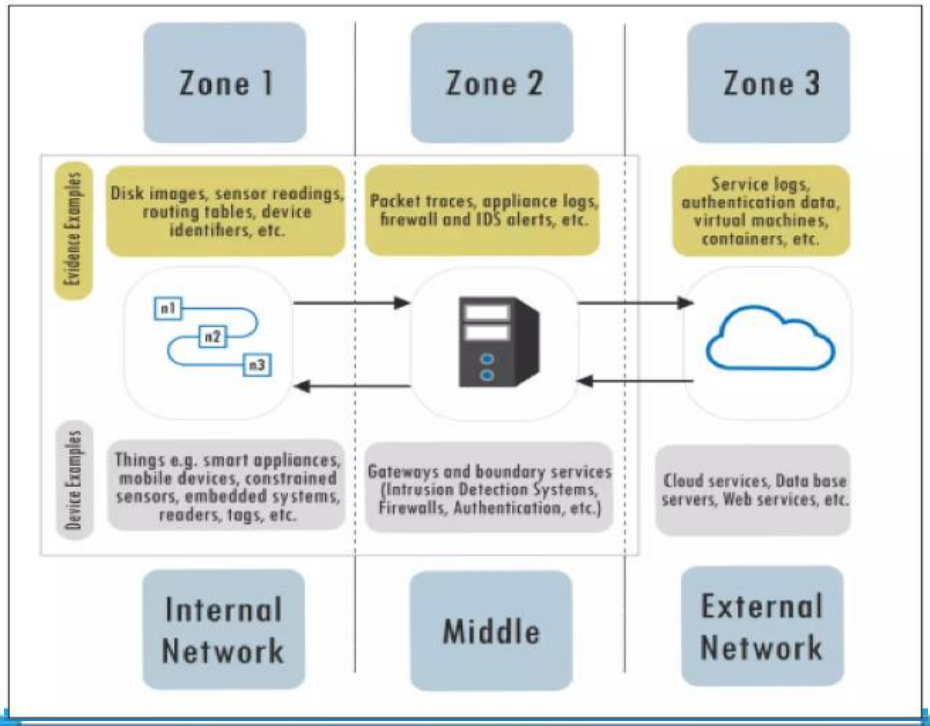
Future Developments



- IoT forensics is a new area open for research. There is already a need for practical solutions to questions that arise during investigations that include IoT.
- Therefore, researchers and forensics professionals work hard to present new tools and methodologies that could mitigate IOT forensic challenges.

The 1-2-3 Zones Approach

- 1-2-3 zones are correspond to three areas of IoT forensics: device, network and cloud.
- This method makes it is easier to plan and systematize an IoT investigation.
- Reduces the complexity and the timing of investigations,
- All zones can be investigate in parallelly or a zone of greatest priority can be investigated first. [6]


Zone 1
Zone 2
Zone 3
Evidence Examples
Disk images, sensor readings, routing tables, device identifiers, etc.
Packet traces, appliance logs, firewall and IDS alerts, etc.
Service logs, authentication data, virtual machines, containers, etc.
n1
n2
n3
Device Examples
Things e.g. smart appliances, mobile devices, constrained sensors, embedded systems, readers, tags, etc.
Gateways and boundary services (Intrusion Detection Systems, Firewalls, Authentication, etc.)
Cloud services, Data base servers, Web services, etc.
Internal Network
Middle
External Network

IOTdots

- IoTDots general architecture divides into two parts:

IoTDots - Modifier (ITM)

- performs source code analysis of smart applications
- Detects relevant forensic information
- The tracing logs are stored in an IoT database

IoTDots - Analyzer (ITA)

- uses the log information stored in the IoT database with data processing and machine learning techniques to perform forensic investigation. [1], [5], [6]

Forensics-aware model for the IoT (FAIoT)

- The FAIoT paper [592] formally defined IoT forensics and listed its challenges.
- Proposed a conceptual model for executing Digital Forensics in the IoT infrastructure. with a centralized trusted evidence repository to ease the process of evidence collection and analysis. [7]
- The FAIoT consists of three parts:
  - secure evidence preservation module.
  - secure provenance module.
  - access to evidence through an API.

Reference

[1]Avoine, G. and Hernandez-Castro,J., n.d. Security of Ubiquitous Computing Systems.

[2]R. C. Joshi and E. S. Pilli, Computer Communications and Networks Fundamentals of Network Forensics A Research Perspective. 2016.

[3]D. Quick and K. K. R. Choo, 'IoT Device Forensics and Data Reduction', IEEE Access, vol. 6, pp. 47566–47574, 2018.

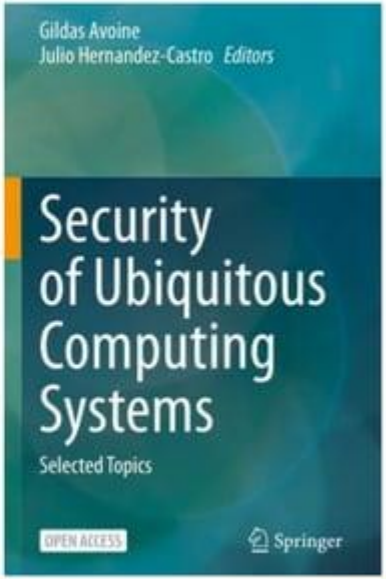
[4]Slideshare.net. 2021. Iot forensics. [online] Available at: <https://www.slideshare.net/AbeisAb/iot-forensics-117926663.

[5]Linkedin.com. 2021. Internet of Things Forensics: Challenges and Approaches. Evaluation of Digital Forensic Tools. [online] Available at: <https://www.linkedin.com/pulse/internet-things-forensics-challenges-approaches-tools-hamal-b-k

[6] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis and E. K. Markakis, "A Survey on the Internet of Things (IoT) Forensics: Challenges, Approaches, and Open Issues," in IEEE Communications Surveys & Tutorials,

[7] S. Zawoad and R. Hasan, 'FAIoT: Towards Building a Forensics Aware Eco System for the Internet of Things',

[8] Hyunji Chung, Jungheum Park, and Sangjin Lee. Digital forensic approaches for amazon alexa ecosystem. Digital Investigation, 22:S15–S25, 2017.



THANK YOU !!!