Cloud, IoT and Enterprise Security

BITS Pilani

Pilani Campus

Nishit Narang
WILPD-CSIS
(nishit.narang@pilani.bits-pilani.ac.in)

IoT Security Framework (Contd.)

Recap: IoT Security Framework

At the heart of the IoT Security Framework are the following key functions:

- Authentication
- Authorization
- Access Control
- (*Apart from the obvious function – Encryption!*)

We discuss these functions in the following slides.

Recap: Authentication

- At the heart of the framework is the authentication layer, used to provide and verify the identity information of an IoT entity
- When connected IoT/ M2M devices (e.g., embedded sensors and actuators or endpoints) need access to the IoT infrastructure, the trust relationship is initiated based on the identity of the device
- The way to store and present identity information may be substantially different for the IoT devices (as against human credentials, like username/password, etc)
  - Device identifiers include RFID, shared secret, X.509 certificates, the MAC address of the endpoint, or some type of immutable hardware based root of trust
  - Establishing identity through X.509 certificates provides a strong authentication system. However, in the IoT domain, many devices may not have enough memory to store a certificate or may not even have the required CPU power to execute the cryptographic operations of validating the X.509 certificates
  - There exists opportunities for further research in defining smaller footprint credential types and less compute-intensive cryptographic constructs and authentication protocols (***aka*** *Lightweight Cryptography*)

Recap: Authorization

- The second layer of this framework is authorization that controls a device's access (to network services, back-end services, data etc)
  - This layer builds upon the core authentication layer by leveraging the identity information of an entity
  - With authentication and authorization components, a trust relationship is established between IoT devices to exchange appropriate information
- Trust relationships can sometimes also be formed in absence of Authorization techniques, and is necessary in some conditions
  - E.g in the absence of a common Authentication and Authorization framework
  - Or, for latency sensitive applications, e.g. those built using the distributed M2M or SIoT architectures

Access Control Types

- Role Based Access Control (or RBAC):
  - Most existing authorization frameworks for computer networks and online services are role based
  - First, the identity of the user is established and then his or her access privileges are determined from the user's role within an organization
  - That applies to most of existing network authorization systems and protocols (RADIUS, LDAP, IPSec, Kerberos, SSH)
- Rule Based Access Control:
  - An administrator may define rules that govern access to a resource
  - Rules may be based on conditions, such as time of day and location
  - Can work in conjunction with RBAC

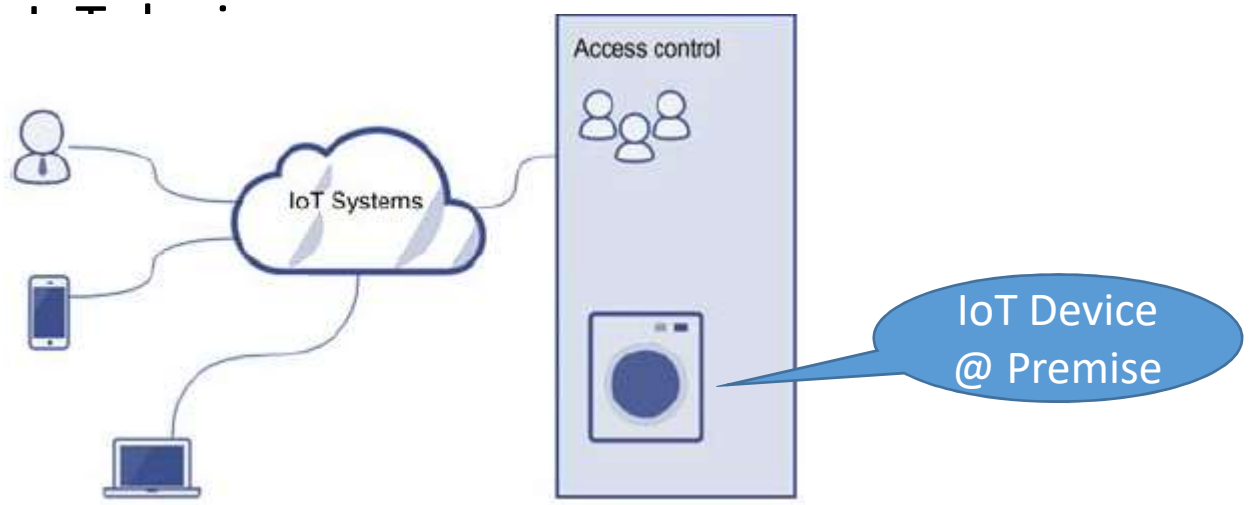
Access Control Types (2)

- Attribute Based Access Control (or ABAC):
  - Attributes (e.g. age, location, etc) are used to allow access
  - Users or devices need to prove their attributes
  - In ABAC, it is not mandatory to verify the identity of the user to establish his or her access privileges, just that the user/device possesses the attributes is sufficient

- Discretionary access control (or DAC):
  - Owners or administrators of the protected system, data or resource set the policies defining who or what is authorized to access the resource
  - Not a good method, since these methods are not centralized and hard to scale

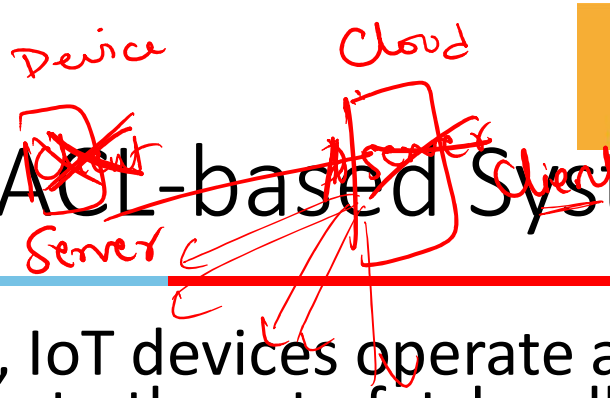
ACL-based Systems

- ACL = Access Control List
- A table that can tell the IoT system all access rights each user/ application has to particular IoT end node
  - Most common privileges include the ability to access or control a



Source: Shancang Li Li Da Xu, Securing the Internet of Things, Syngress, 1st Edition, 2017

Challenge with ACL-based Systems



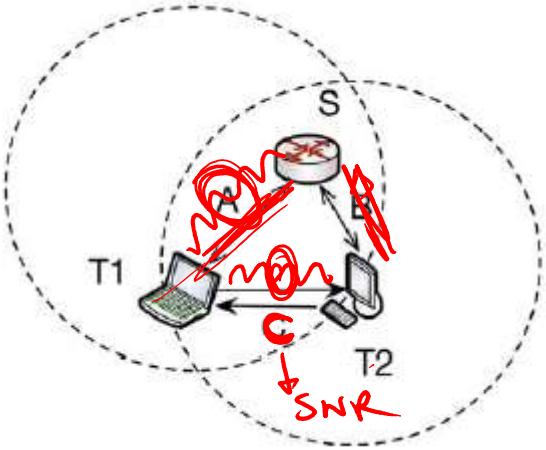
- In many architectures, IoT devices operate as “servers”, with clients connecting to them to fetch collected data
- Server IP and port information is public knowledge => no security
- Minimum security is typically implemented using <username, password> → an embodiment of IoT ACL-based device systems
  - Approach is not scalable as more users join or are revoked
  - Complexity of managing the ACL at the device can become a bottle-neck
- A more scalable approach for IoT is to use “capabilities” for enabling “**capability-based access**”
  - A capability is essentially a cryptographic key, that gives access to some ability (e.g. to communicate with the device)

IoT Security: Implementation Methods

Lightweight Cryptography

- Recently, the lightweight cryptography for IoT has attracted lots of research effort
  - The traditional cryptography is designed at the application layer without regard to the limitations of IoT Devices, making it difficult to directly apply the existing cryptography primitives to IoT
- Recently, the idea of designing lower layer security schemes, such as physical layer crypto and lightweight crypto supports the resources (computation, RAM, energy supply, etc.) limited to IoT devices

- The idea of physical layer security scheme and lightweight cryptography over resource-limited IoT devices first appeared in the works of Wyner (1975) and Korner (2002)
  - They investigated a channel model using the "***wiretap channel***," in which a transceiver attempts to communicate reliably and securely with a legitimate receiver over a noisy channel, while its messages are being eavesdropped by a passive adversary through another noisy channel



Source: Shancang Li Li Da Xu, Securing the Internet of Things, Syngress, 1st Edition, 2017

Source: https://www.cloudflare.com/en-in/learning/security/glossary/iot-security/

Lightweight Cryptography: Some Background

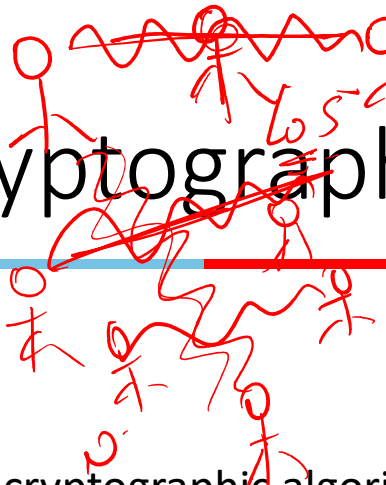
- **Definition**: Information-theoretic Security
  - *"A cryptosystem is considered to have information-theoretic security (also called **unconditional security**) if the system is secure against adversaries with unlimited computing resources and time. In contrast, a system which depends on the computational cost of cryptanalysis to be secure (and thus can be broken by an attack with unlimited computation) is called computationally, or conditionally, secure."*
- The concept of information-theoretic secure communication was introduced in 1949 by American mathematician Claude Shannon, one of the founders of classical information theory
  - In Shannon's wiretap model, he assumed both the main and eavesdropper's channels to be noiseless
  - Shannon's results discouraged further research in information theoretic secrecy
- Wyner revisited this problem with relaxed assumptions, mainly:
  - The noiseless communication assumption of Shannon was relaxed by assuming a possibly noisy main channel and an eavesdropper channel that is a noisy version of the signal received at the legitimate receiver
  - Wyner's results showed that positive secure rates of communication are achievable, under certain conditions of noise or interference in the channels

Source: Wikipedia

Lightweight Cryptography: Some Background (2)

- Wyner's work is highly applicable to Wireless Channels (most often used for IoT communications)
  - Wireless channels have natural impairments (e.g. attenuation in signal strength) that make the received signal different from the one originally transmitted
  - Secure communication without the need to share a secret key, or what is now called as the ***key-less security approach*** suggested a new paradigm of secure communication protocols
  - That is, exploiting properties of the wireless medium (noise or interference or jamming) to satisfy the secrecy constraints
- The key-less security approach can be used in wireless networks to securely exchange the shared-secret key between two communicating nodes, which can be used for all subsequent communications
  - Simpler alternative to secure key exchange protocols like the Diffie-Hellman Key Exchange (which is based on Mathematical Principles of Elliptic Curve Cryptography)

Lightweight Cryptographic Algorithms



- Lightweight cryptography is a cryptographic algorithm or protocol tailored for implementation in constrained environments including RFID tags, sensors, contactless smart cards, healthcare devices, and so on.
- Due to IoT push, Lightweight cryptography has gained momentum:
  - The properties of lightweight cryptography have already been discussed in ISO/IEC 29192 in ISO/IEC JTC 1/SC 27
    - ISO/IEC 29192 is a new standardization project of lightweight cryptography, and the project is in process of standardization
  - NIST received 57 submissions to be considered for standardization. After the initial review of the submissions, 56 were selected as Round 1 Candidates
  - Due to the large number of submissions and the short timeline of the NIST lightweight cryptography standardization process, some of the candidates were eliminated from consideration early in the first evaluation phase in order to focus analysis on the more promising ones
    - Out of 56 Round 1 candidates, 32 were selected to go for Round 2 evaluation
  - On March 29, 2021, NIST announced ten finalists as ASCON, Elephant, GIFT-COFB, Grain128-AEAD, ISAP, Photon-Beetle, Romulus, Sparkle, TinyJambu, and Xoodyak
  - The final round of the standardization process was expected to last approximately 12 months. NIST will host a workshop near the end of the evaluation period

Transport Encryption

- SSL/TLS, DTLS (Chapter 4)
- The transport encryption is done using secure transport protocols such as TLS and SSL
  - Both the TLS and SSL are cryptographic protocols that provide communications security over a network
  - TLS uses TCP and therefore does not encounter packet reordering and packet loss issues
- Datagram Transport Layer Security (DTLS):
  - DTLS is developed based on TLS by providing equivalent security services, such as confidentiality, authentication, and integrity protection
  - In DTLS, a handshake mechanism is designed to deal with the packet loss, reordering, and retransmission
  - DTLS provides three types of authentication:
    - non-authentication, server authentication, and server and client authentication

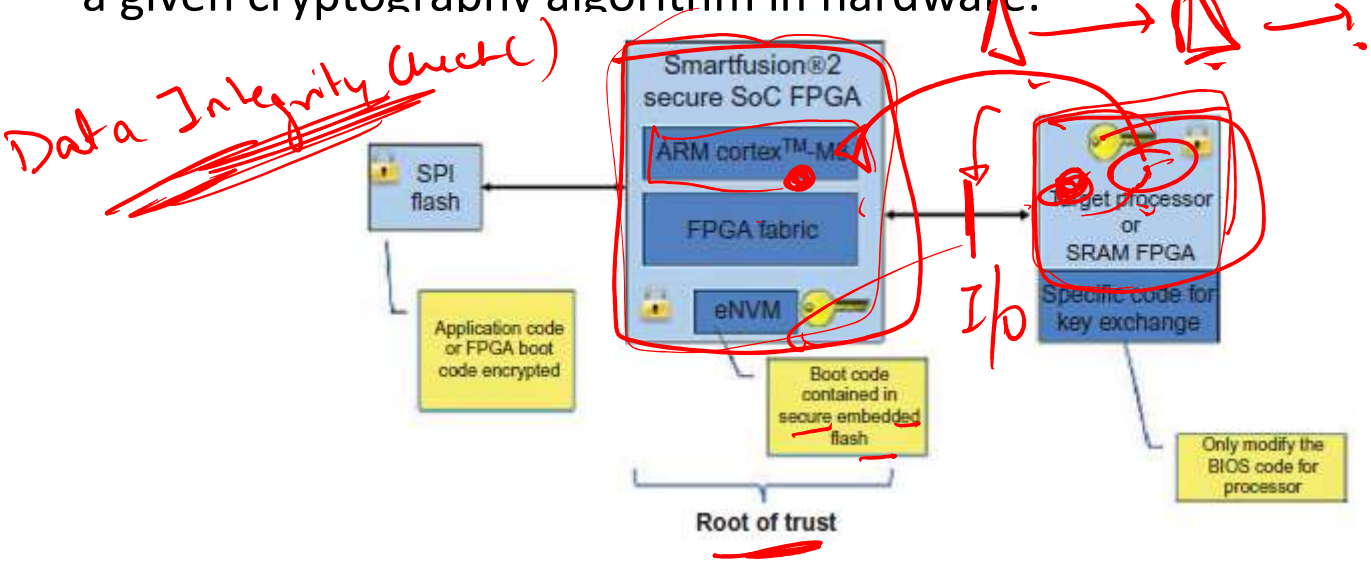
mutual TLS (mTLS)

- Mutual Transport Layer Security (mTLS) is a type of mutual authentication, which is when both sides of a network connection authenticate each other.
  - TLS is a protocol for verifying the server in a client-server connection;
  - mTLS verifies both connected devices, instead of just one.
- mTLS is important for IoT security because it ensures only legitimate devices and servers can send commands or request data.
  - It also encrypts all communications over the network so that attackers cannot intercept them.
- mTLS requires issuing TLS certificates to all authenticated devices and servers.
  - A TLS certificate contains the device's public key and information about who issued the certificate.
  - Showing a TLS certificate to initiate a network connection can be compared to a person showing their ID card to prove their identity.

Source: https://www.cloudflare.com/en-in/learning/security/glossary/iot-security/

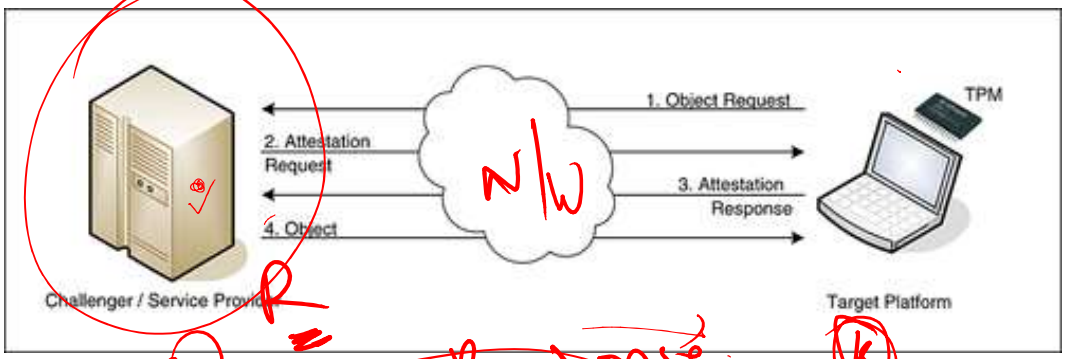
Hardware Security Solutions

- Hardware approach security is a secure way to protect IoT devices, which can use hardware chips (such as application-specific integrated circuits (ASICs) or field-programmable gate array (FPGA)) to implement a given cryptography algorithm in hardware.

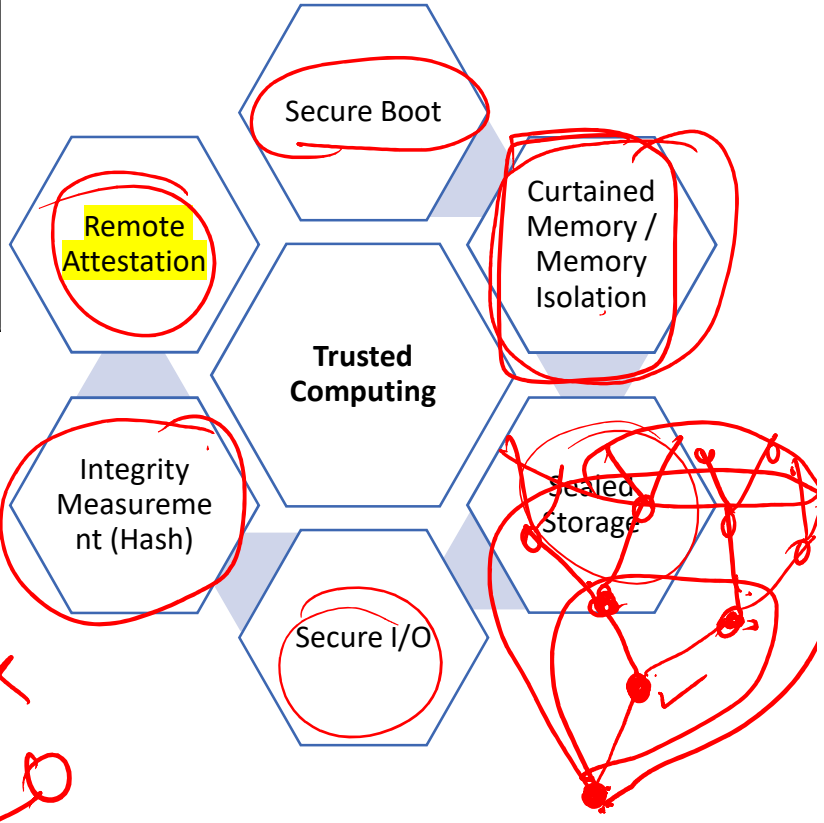


Secure boot: It is a process involving cryptography that allows an electronic device to start executing authenticated and trusted software to operate.
It is the foundation of trust but the nodes still need protection from various run-time threats.

Device Attestation Techniques



Source: https://doi.org/10.1002/sec.393



- **Attestation techniques**
- HW-based Solution (TPM)
- SW-based Solution (e.g. PIONEER, 2005)
- Hybrid Solution (e,g. SMART – Secure & Minimal Architecture for Remote Trust)
- Swarm Attestation (e.g. SEDA, SANA..)

Security Analytics

- This method can be used for detection of compromised devices
- A security analytics infrastructure can significantly reduce vulnerabilities and security issues related to the Internet of Things
  - This requires collecting, compiling, and analyzing data from multiple IoT sources, combining it with threat intelligence, and sending it to the security operations center (SOC)
  - Applies AI/ML practices to IoT Security

- Extra Reading:
  - An analytics framework to detect compromised IoT devices using mobility behavior: DOI: 10.1109/ICTC.2013.6675302

Use of Raw Public Keys (RPKs)

- In resource-constrained IoT devices, such as intelligent sensors or RFID tag, the certificate chains or even single certificate may be too big to process
- Recently, IETF recommended the use of RPKs instead of certificates for TLS and DTLS
  - IETF RFC 7250 *[Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)]*
- Traditionally, TLS client and server public keys are obtained in PKIX containers **in-band** as part of the TLS handshake procedure and are validated using trust anchors based on a [PKIX] certification authority (CA)
  - TLS is, however, also commonly used with self-signed certificates in smaller deployments where the self-signed certificates are distributed to all involved protocol endpoints out-of-band.
  - This practice does, however, still require the overhead of the certificate generation even though none of the information found in the certificate is actually used

Use of Raw Public Keys (RPKs) [2]

- Alternative methods are available that allow a TLS client/server to obtain the TLS server/client public key:
  - The TLS client can obtain the TLS server public key from a DNSSEC-secured resource record using DNS-Based Authentication of Named Entities (DANE) [RFC6698]
  - The TLS client or server public key is obtained from a [PKIX] certificate chain from a Lightweight Directory Access Protocol [LDAP] server or web page.
  - The TLS client and server public key is provisioned into the operating system firmware image and updated via software updates.
- With raw public keys, only a subset of the information found in typical certificates is utilized: namely, the *SubjectPublicKeyInfo* structure of a PKIX certificate that carries the parameters necessary to describe the public key
  - Public-Key Information of a certificate carries the public key values and the algorithm identifier of the cryptographic algorithm used to generate it. Other parameters found in PKIX certificates are omitted
  - This results in the raw public key being fairly small in comparison to the original certificate, and the code to process the keys can be simpler
  - Only a minimalistic ASN.1 parser is needed; code for certificate path validation and other PKIX-related processing is not required
- The RPK requires the out-of-band validation of the public key
  - The mechanism defined in the RFC only provides authentication when an out-of-band mechanism is also used to bind the public key to the entity presenting the key

innovate achieve lead

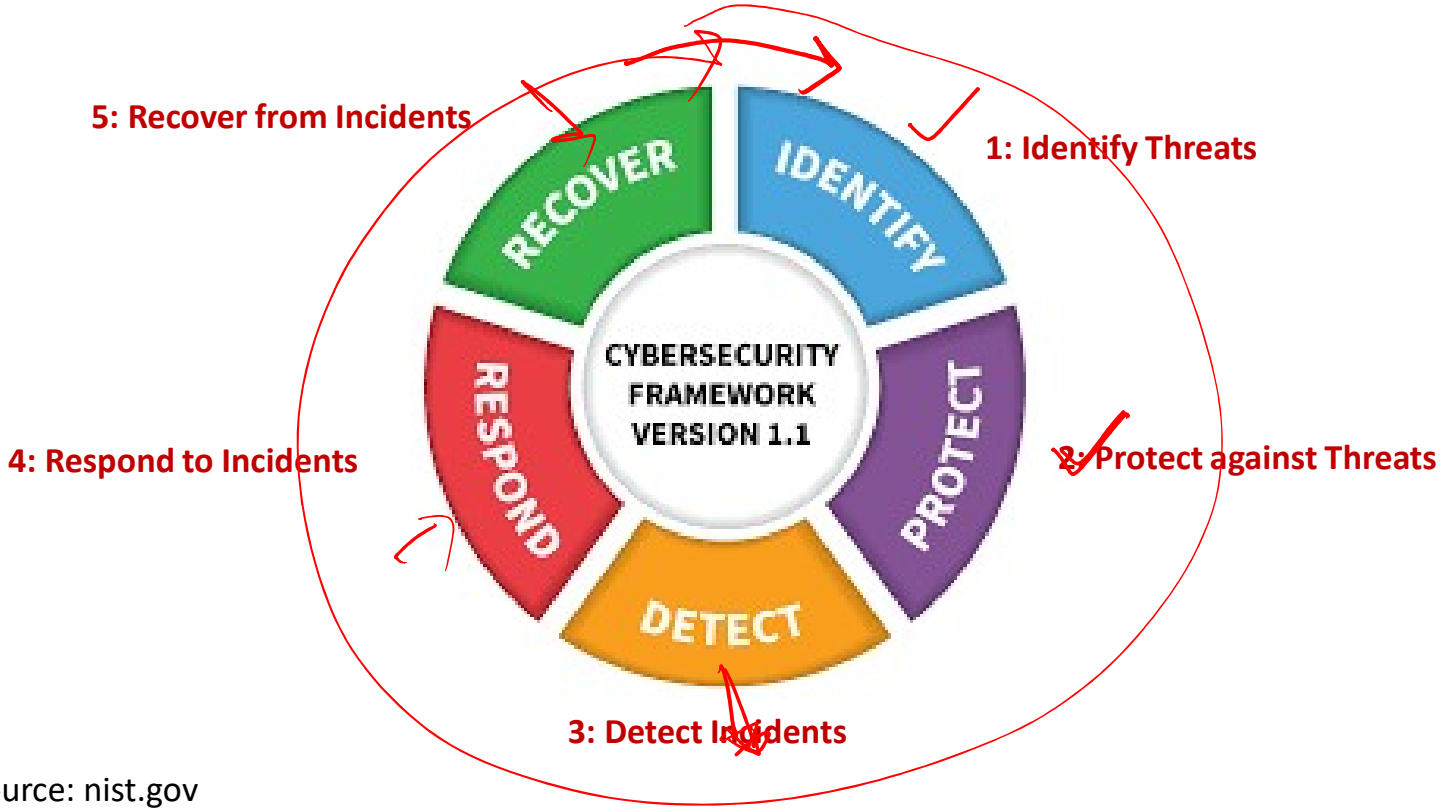
BITS Pilani

Pilani Campus

IoT Security

Sub-Topic: IoT Forensics

Cybersecurity: NIST Framework



Source: nist.gov

What is Cyber Forensics….

- "…. is the application of investigation and analysis techniques to gather and preserve evidence from a particular computing device in a way that is suitable for presentation in a court of law."

Source: What is Computer Forensics (Cyber Forensics)?
https://searchsecurity.techtarget.com ›

Definition: Digital Forensic Science (DFS)

"The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations."

Source: (2001). Digital Forensic Research Workshop (DFRWS)

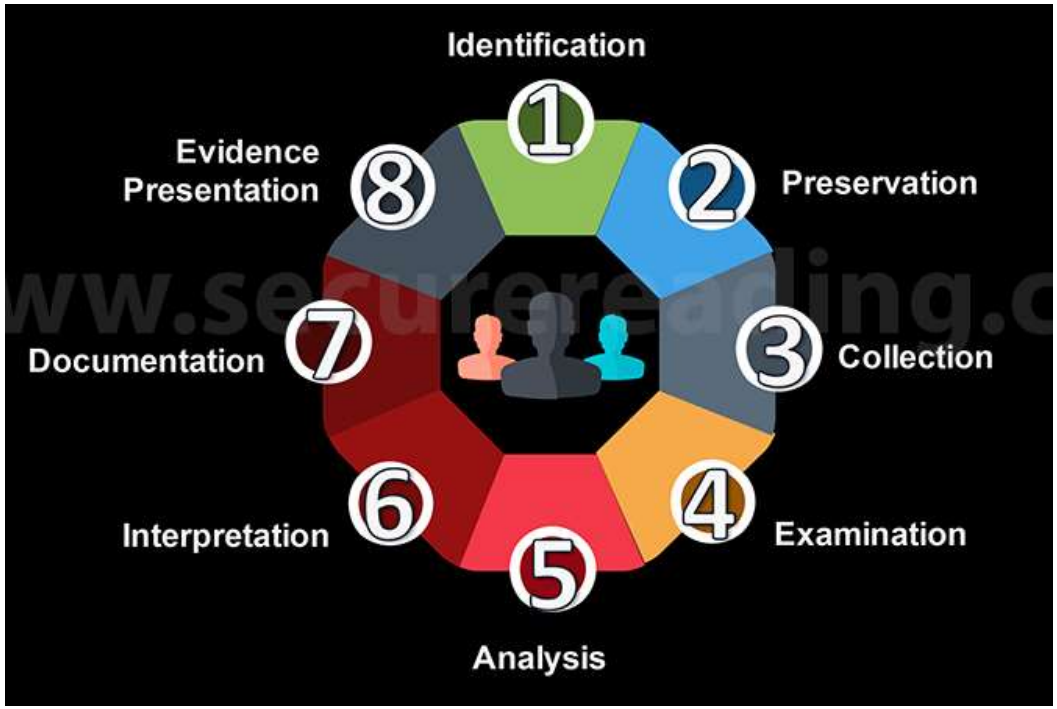
Digital Forensic Science

Source: Cyber Forensics by Eric Katz

Table 1 - Suitability Guidelines for Digital Forensic Research

Area	Primary Objective	Secondary Objective	Environment
Law Enforcement	Prosecution		After the fact
Military IW Operations	Continuity of Operations	Prosecution	Real Time
Business & Industry	Availability of Service	Prosecution	Real Time

Digital Forensics Process



Source: https://secureReading.com/digital-forensics/

History of Cyber Forensics

Computer Forensics

- Focused on computing devices
- Requires understanding of Boot Process, File Systems, Registry / Configuration Files, OS Functions etc.
- Static and Live Acquisition of Data

Network Forensics

- Systematic tracking of incoming and outgoing traffic to ascertain how an attack was carried out
- Determine the cause of abnormal traffic (internal bug, attackers)
- Live acquisitions are especially useful

Data Storage Devices

- Requires understanding of data storage devices for data acquisition
- SCSI Disks, IDE/EIDE Disks, SATA Drives
- CD, CD-R, CD-RW, DVD
- RAID Systems (RAID 0, 1, 2, 3, 4, 5, 6, 10)

Evolution of Cyber Forensics

Mobile Device Forensics

- A wealth of information on cell phones/ Smart Phones
- Crimes targeting Mobile Devices
- Requires understanding of Mobile Device Organization, OS, File System and Storage system

Cloud Forensics

- Combines cloud computing with digital forensics
- Requires investigators to work with multiple computing assets, such as virtual and physical servers, networks, storage devices, applications, and much more

IoT Forensics

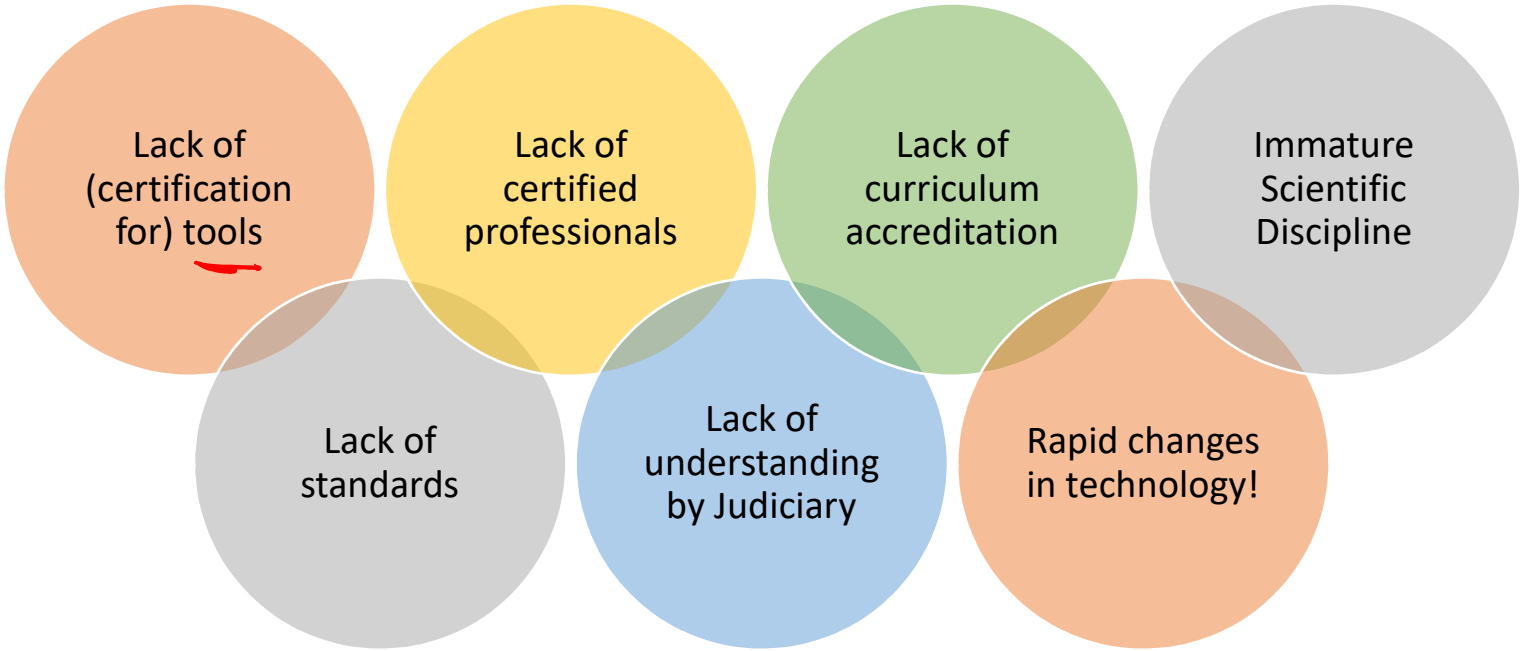
- IoT is a combination of many technology zones: Device, Network and Cloud
- IoT Forensics thus covers: Cloud forensics, Network forensics and Device forensics.
- Evidence could be from home appliances, cars, tags readers, sensor nodes, medical implants in humans or animals, or other IoT devices

Introduction to IoT Forensics

- Growth in IoT raises challenges for the digital investigator when IoT devices involve in criminal scenes
- Current research in the literature focuses on security and privacy for IoT environments rather than methods or techniques of forensic acquisition and analysis for IoT devices
- Cybercrimes with the power of IoT technology can cross the virtual space to threaten human life and the increasing number of these crimes is one of the main reasons why we need IoT forensics
- IoT digital evidence is a rich and often unexplored source of information
- From the forensic perspective, each IoT device will provide important artifacts that could help in the investigation process

Source: https://arxiv.org/abs/1801.10391

IoT Forensics: Issues



Devices Identification: A Complex Task with IoT!












Source: Cyber Forensics by Eric Katz

Traditional Forensics Vs IoT Forensics

There are several aspects of differences and similarity between traditional and IoT forensics

- In terms of evidence sources, traditional evidence could be computers, mobile devices, servers or gateways. In IoT forensics, the evidence could be home appliances, cars, tags readers, sensor nodes, medical implants in humans or animals, or other IoT devices.
- In terms of Jurisdiction and Ownership, there are no differences, it could be individuals, groups, companies, governments, etc.
- In terms of evidences data types, IoT data type could be any possible format, it could be a proprietary format for a particular vendor. However, in traditional forensics, data types are mostly electronic documents or standard file formats.
- In terms of networks, the network boundaries are not as clear as the traditional networks, increasing in the blurry boundary lines.

Source: https://arxiv.org/abs/1801.10391

IoT Forensics

- IoT technology is a combination of many technology zones: IoT zone, Network zone and Cloud zone.
- These zones can be the source of IoT Digital Evidences
- Evidence can be collected from a smart IoT device or a sensor, from an internal network such as a firewall or a router, or from outside networks such as Cloud or an application.
- Based on these zones, IoT Forensics covers three aspects in term of forensics: Cloud forensics, network forensics and device level forensics.
  - Most of IoT devices have the ability to (directly or indirectly) connect through applications to share their resources in the Cloud, with all valuable data that is stored in the Cloud → Cloud Forensics.
  - Different kinds of networks that IoT devices use to send and receive data. It could be home networks, industrial networks, LANs, MANs and WANs. For instance, if an incident occurs in IoT devices, all logs from network devices through which the traffic flows could be potential evidence
  - Device Level Forensics include all potential digital evidence that can be collected from IoT devices like graphics, audio, video. Videos and graphics from CCTV camera or audios from Amazon Echo, can be great examples of digital evidences in the device level forensics.

Source: https://arxiv.org/abs/1801.10391

Challenges in IoT Forensics

- Data Location:
  - Most of the IoT data is spread in different locations, which are out of the user control. This data could be in the Cloud, in third party's location, in mobile phone or other devices.
  - To identify the location of evidence is considered as one of the biggest challenges an investigator can face in order to collect the evidence.
  - In addition, IoT data might be located in different countries and be mixed with other users information, which means different countries regulations are involved
- Lifespan limitation of Digital Media Storage:
  - Because of limited storage in IoT devices, the lifespan of data in IoT devices is short and data can be easily overwritten, resulting in the possibility of evidence being lost
  - Therefore, one of the challenges is the period of survival of the evidence in IoT devices before it is overwritten.
  - Transferring the data to a local Hub or to the Cloud could be an easy solution to solve this challenge. However, it presents challenges related to securing the chain of evidence and to prove the evidence has not been changed or modified
- Lack of Individual Identity:
  - Even though the investigators find an evidence in the Cloud that prove a particular IoT device in crime scene is the cause of the crime, it does not mean this evidence could lead to identification of the criminal

Source: https://arxiv.org/abs/1801.10391

Challenges in IoT Forensics (Contd.)

- Lack of Security:
  - Evidence in IoT devices could be changed or deleted because of lack of security, which could make these evidence not solid enough to be accepted in a court of law
- Variety of Device Types:
  - In identification phase of forensics, the digital investigator needs to identify and acquire the evidence from a digital crime scene.
  - Usually, evidence source is types of a computing system such as computer and/or a mobile phone.
  - However, in IoT, the source of evidence could be objects like a smart refrigerator or smart coffee maker
  - The device could be turned-off because it could have run out of battery, which makes its chances to be found difficult, especially if the IoT devices is very small, in hidden places or looks like a traditional device.
  - Carrying the device to the lab and finding a space could be another challenge that investigators face
  - Extracting evidence from these devices is considered another challenge as most of the manufacturers adopt different platforms, operating systems and hardware.
- Lifecycle Changes in Data Formats:
  - The format of the data that is generated by IoT devices is not identical to what is saved in the Cloud.
  - Data processing using analytic and translation functions in different places is likely before being stored in the Cloud. Hence, in order to be accepted in a court of law, the data form should be returned to its original format before performing analysis

Source: https://arxiv.org/abs/1801.10391