Mitigating Cyber Attacks on IoMT Devices

ISM 6124 – Advanced Information Systems Analysis and Design
Fall Semester 2018
Hot Topic Report

Prepared for

Professor Alan R. Hevner
University of South Florida

Prepared by

Srirag Ramadugu
U23695579

October 25, 2018

Table of contents:

Introduction:

Digital health, which includes the use of connected health devices in both clinical and non-clinical settings, bids opportunities to create economic and social benefits by transforming social care and health best practices. With the growing expectations of healthcare, a transformation is necessary as global healthcare costs are rising faster than the global economy. Air pollution and an ageing population in addition to the increasing chronic disease burden has been worsened.

IoT, smartphones and modern software tools can support clinical decisions and allow patients to be managed and treated remotely. Connected health devices, a key set of tools, range in scale and complexity from implantable devices such as cardiac pacemakers, drug administration devices and monitoring devices to non-implantable devices such as infusion pumps, defibrillators, glucometers and blood pressure measurement devices. Connected health devices also include large-scale hospital equipment such as MRI scanners and x-ray machines.

Connected health devices are part of an international supply chain and to enhance the prospects for digital health there is a multidisciplinary expertise such as health, data transfer, engineering. A process is needed to map the scale of impact from cyberattacks against the range of applications.

1.Layout

In recent years a wide range of wearable IoT healthcare devices have been developed. These devices allow the transfer of patient personal information between different devices, at the same time personal health and wellness information of patients can be tracked and attacked.

"'The Big Data Bang' is an IoT world that will explode from 2 billion objects (smart devices which communicate wirelessly) in 2006 to a projected 200 billion by 2020", according to Intel. Millions of people can be hacked and digitally monitored their implantable medical devices which include cardioverter defibrillators, pacemakers, deep brain neurostimulators, insulin pumps, ear tubes, and more.

Medical device classes:

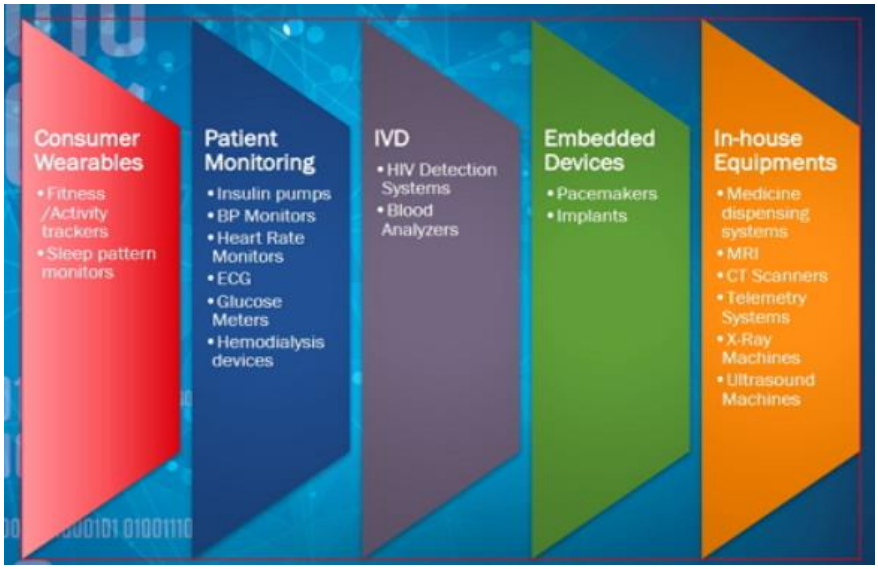
According to Lake et al., "the chances of the chances of security breaches increase in direct proportion to the "degree of connectivity". The number of medical devices that are internet enabled has been drastically improved. These devices allow attackers to obtain sensitive information and infect devices with malware, thus endangering human lives. Devices like neurostimulators, implantable cardiac defibrillators (ICDs), pacemakers and drug delivery systems are the primary targets for the cyber-attacks. Before devices being released into the market, Food and Drug Administration (FDA) set up some cybersecurity guidelines for three medical devices classes as per below table.

Medical Device Class	Attributes	Example Devices
Class I	Common, low risk, low complexity	Lancet, Dental Floss
Class II	More complex, greater risk to patient, partially implanted	Syringe, Insulin Pump, BGM
Class III	Fully implanted, greater risk, regulate body functions	Artificial Pancreas, CGM, Replacement Heart Valves

2.Medical device class

As per FDA, class one medical devices like Lancet and Dental Floss have low risk, low complexity and cybersecurity concerns generally revolve around class two and three devices which include complexity, greater risk to patients, that are partially and fully implanted and that which regulate body functions. Specially, with class three devices like artificial pancreas, CGM, Replacement heart valves being affected, human life would be at great risk. A classic example for this would be, Jay Radcliffe, who hacked into an insulin pump, a class two device and could send the command to disable the device. For diabetics who are reliant on properly functioning insulin pumps, this is devastating.

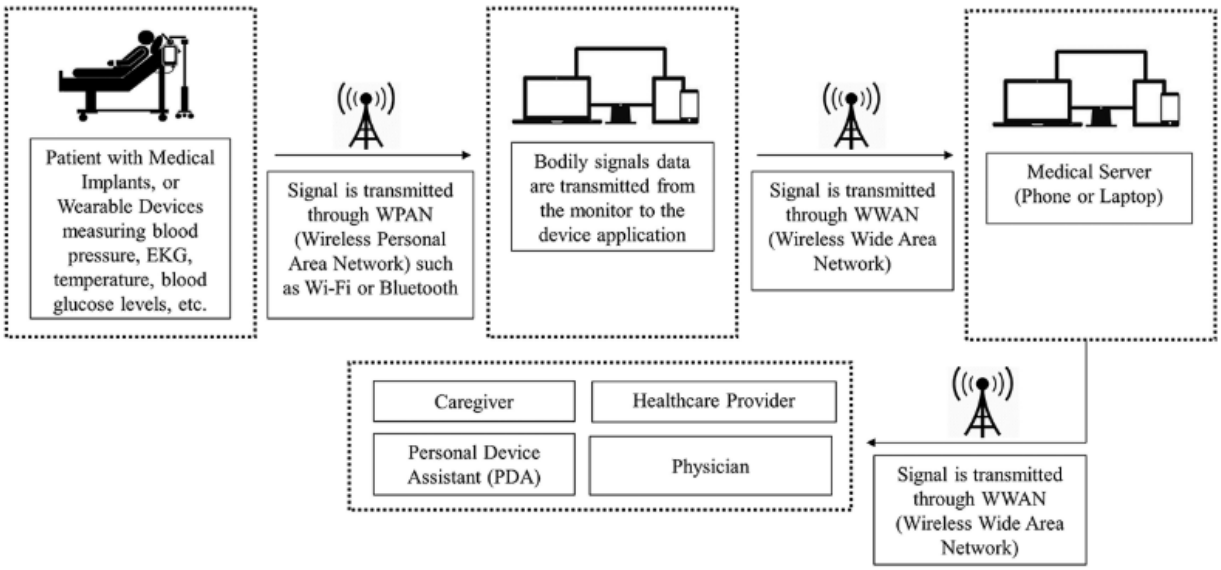
Medical Device Categories:



3.Medical device categories

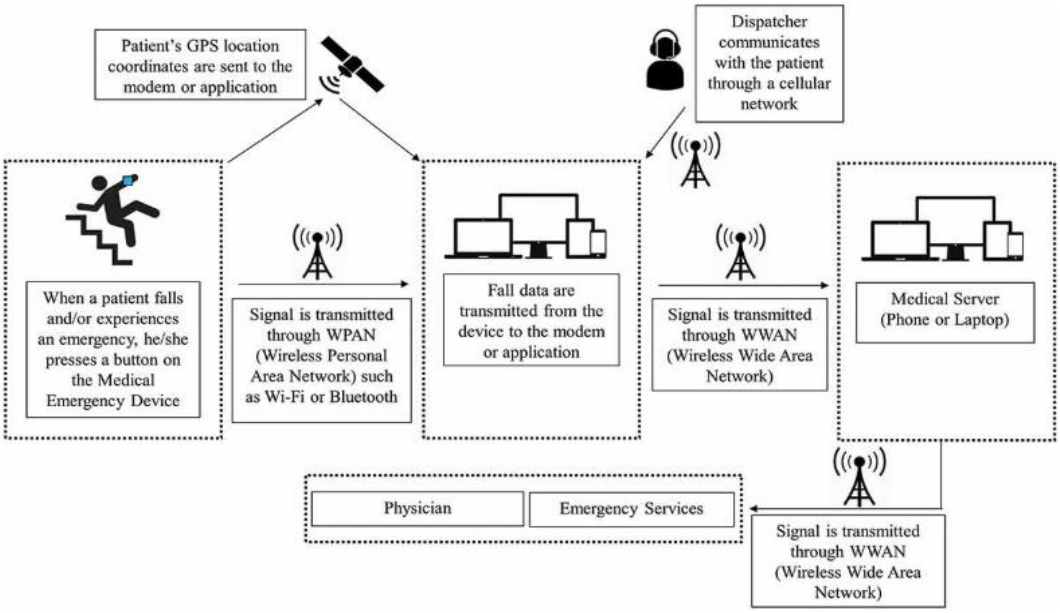
Wireless Implantable and Wearable Devices:

The below figure shows the patient's electrical heart activity that is captured through an EKG in Pennsylvania. It transfers information, to a device via Bluetooth and foster to a data center Via Wi-Fi. The physician, in California, is alerted through SMS and can log into hospital system to look into the patient reading. Implantable and wearable devices live examples include Medtronic, MiracleEar, Dexcom, Insulet, Fitbit and Dario.



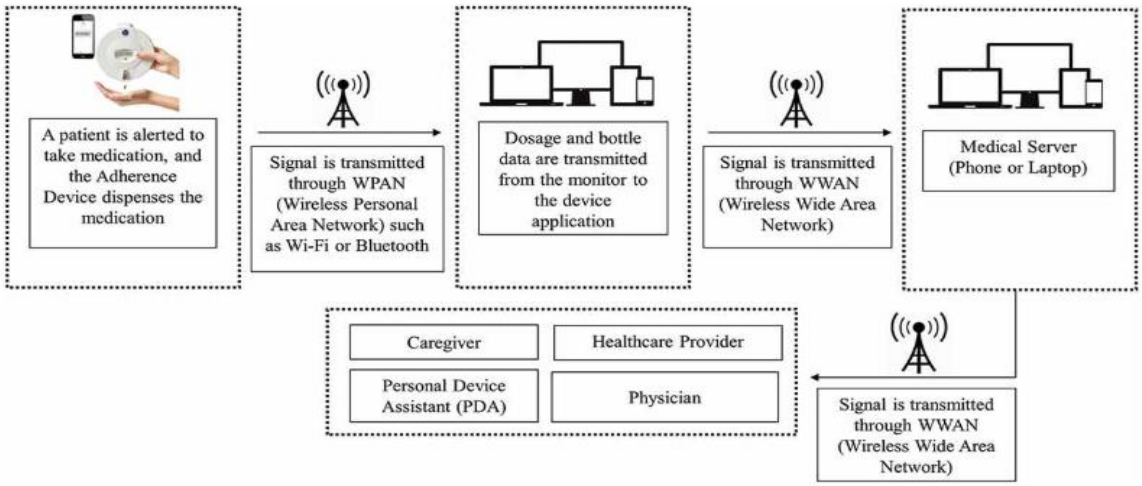
Wireless Emergency Response Devices:

The device will capture its GPS location and patient abnormal activities. These signals are communicated to the healthcare server and to the emergency medical services. Emergency response devices live examples include bay alarm medical, medical guardian, alert 1, qmedic and life alert.



Wireless Medical Adherence Devices:

A medication dispenser alerts a remote server through internet. The server sends the medication information to a smart phone application. The remote server sends a command back to the dispenser to close the valve the user of completion.



Vulnerability Assessment Approaches:

Vulnerability Assessment aims to provide organizations with knowledge of systems susceptible to cyber-attacks. Steps to conducting a vulnerability assessment:

1. Define assessment scope
2. Utilize software to identify vulnerabilities
3. Analyze the software-generated reports
4. Attempt to exploit the system using the known vulnerabilities

Previously, the above procedures were used to assess the vulnerability on SCADA (Supervisory control and data acquisition) systems and generic IOT systems. Today, there are many vulnerability assessment tools such as Burp Suite, Nessus, Qualys, Nexpose, etc.,

These tools offer coverage of over 1000 generic vulnerabilities, such as SQL injection and cross-site scripting (XSS), their Infiltrator technology can be used to perform interactive application security testing (IAST) by instrumenting target applications to give real-time feedback to their Scanner when its payloads reach dangerous APIs within the application. They can perform scheduled scans at specific times or carry out one-off scans on demand. They can configure repeat scans to run indefinitely or until a defined end and can view in a single place the entire scan history for a given web site.

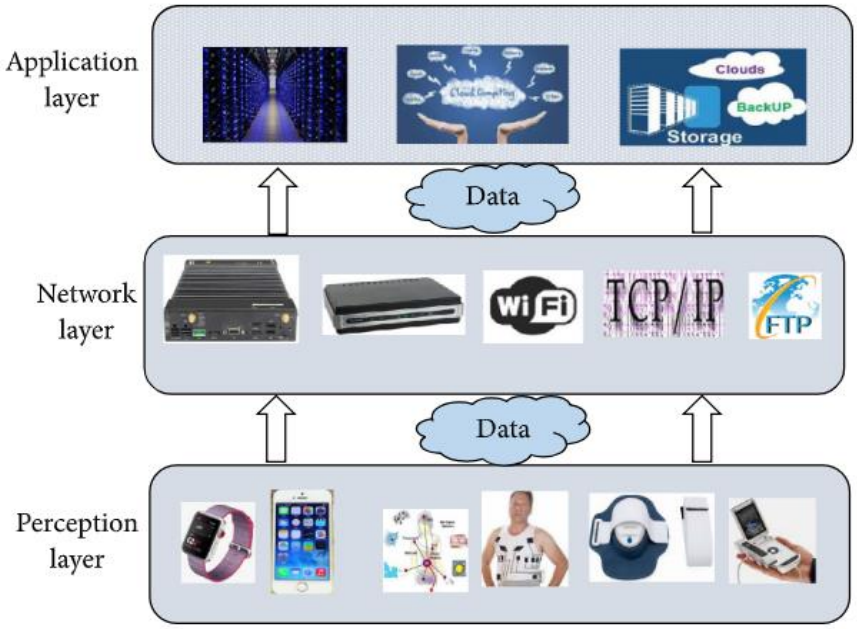
Among the above listed vulnerability assessment tools, Nessus has been identified as the ideal one for large-scale vulnerability assessments. It offers the most diverse set of plugins for assessing a broad range of technology such as SCADA devices, web applications, and Window/Linux systems. Nessus was designed to scan networks with thousands of devices. It categorizes vulnerabilities into five risk categories ('Critical', 'High', 'Medium', 'Low', and 'None') based on the industry standard Common Vulnerability Scoring System (CVSS).

Structure of IoMT (Internet of Medical Things):

Medical Internet of Things has provided great potential in providing better medical care for the people by supporting applications from simple medical devices to Wireless Body Area Network. Medical Internet of Things structure is composed of three layers:

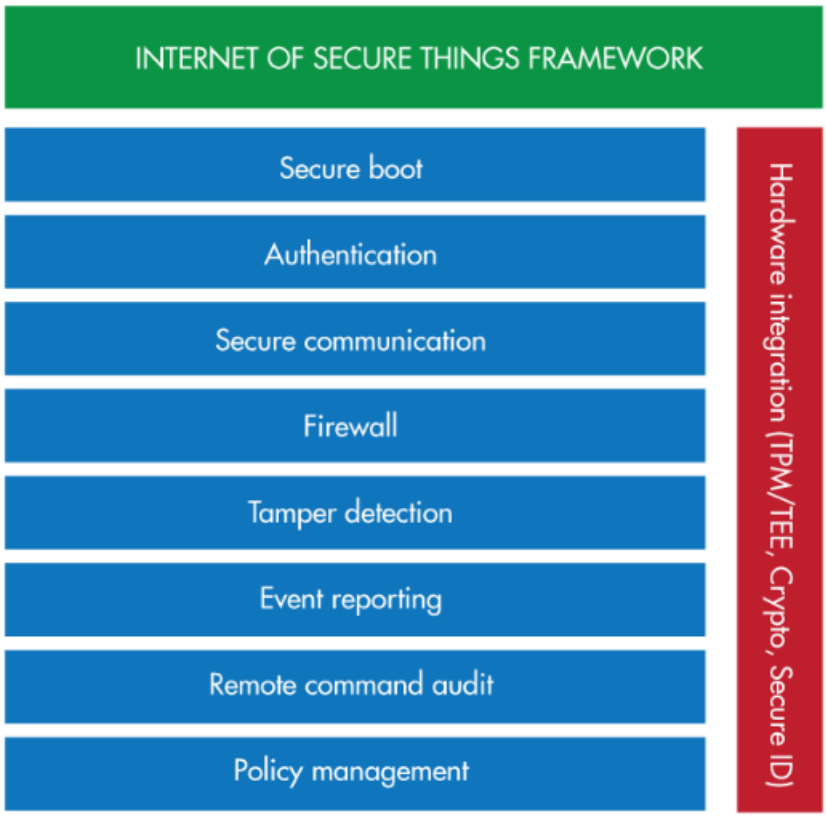
1. The perception layers
2. The network layers
3. The application layers

The perception layer is supported by many technical platforms and from which the network layer processes and transmits the data wired or wirelessly. It is up to the network layers well developed transport protocols that will decide the transmission efficiency and energy consumption. The application layer on the other hand, integrates the medical information resources to provide the personalized medical services and thereby provide better service to the customers.



Security Framework:

A security solution for IoT devices must provide protection against a wide range of cyber-attacks. It must ensure that the device has not been tampered and should be able to secure the data stored by the device. It must secure both inbound and outbound communications, detect and report attempted cyber-attacks. This is possible only by adopting secure features right from the beginning.



Even though there is no perfect security solution for any embedded device, there are some solutions that are available which provide framework for OEMs (Original equipment manufacturers). Such security framework will help the original equipment manufacturers in including core capabilities that are required for the protection of their devices and to know the flexibility that they must customize the solution to the specific requirements of their device. In doing all these they can manage to keep security features in track using such frameworks.

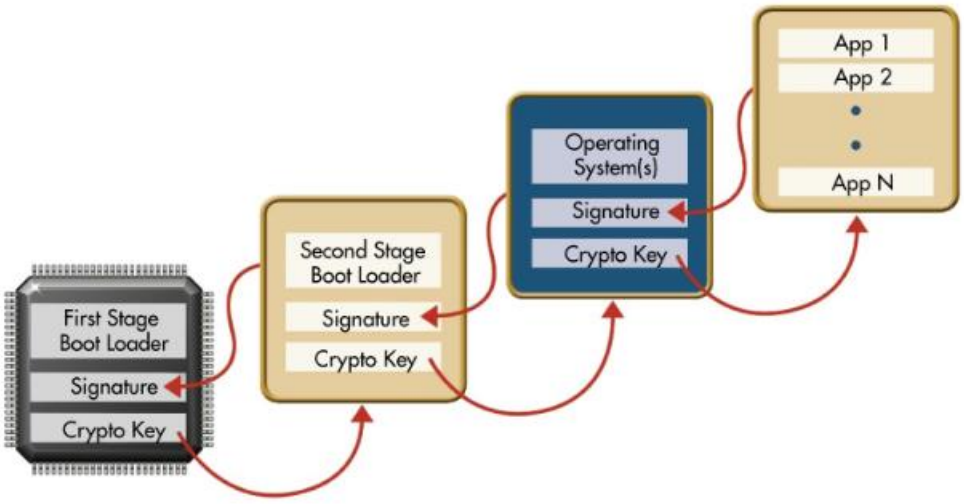
Secure Boot:

The biggest security threat is when IoT device installs malware or code modified by the hacker. This can be curbed by using secure boot.

Secure boot begins with first stage bootloader which is programmed into a protected and non-writable storage location on the device. First stage boot loader checks the authenticity of the second stage boot loader. The second stage bootloader which is stored in reprogrammable flash drive and which is way complex than first, repeats the same process i.e., verifying the authenticity.

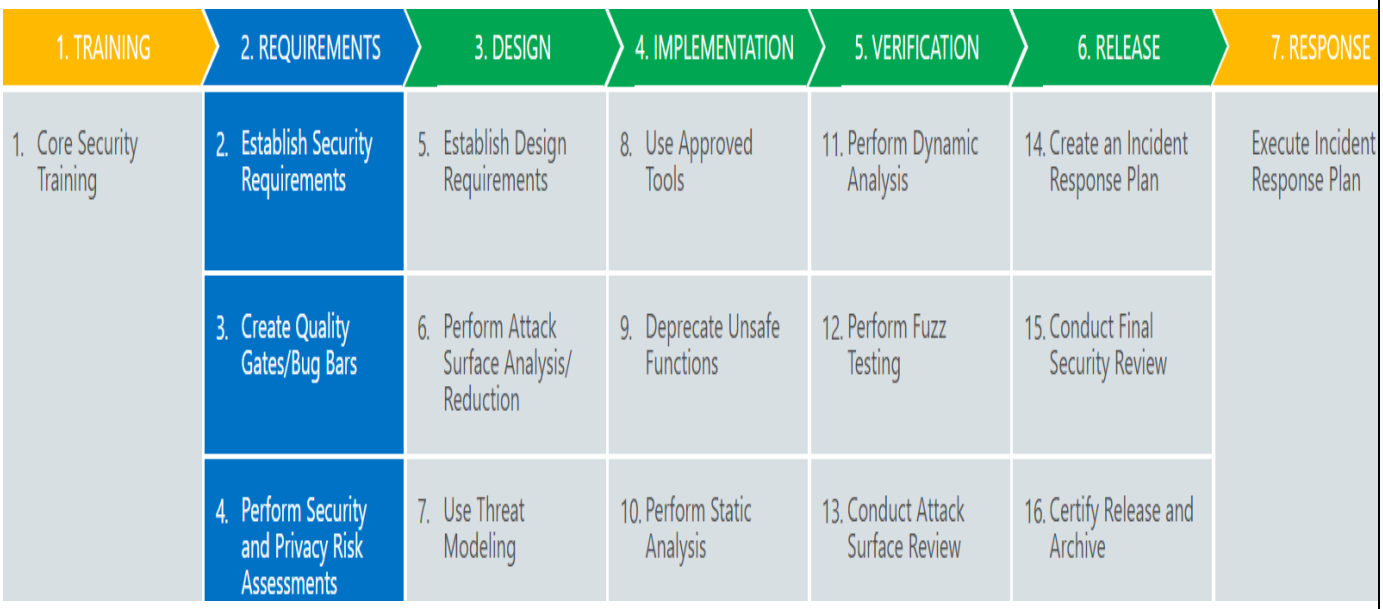
Secure boot relies on the signed code images signed by Original equipment manufacturers using their private key. When an update is sent, the device uses its public key to validate the signed image and thereby detects the malicious content and discards it. Thus, only valid images are accepted and saved to the devices.

From the below diagram, it is clear how first stage boot loader checks the authenticity of second stage boot loader by checking its signature image and thereby validating crypto keys and thereby making secure boot possible.



Life Cycle:

In the security life cycle of Microsoft, the project requirements phase is the best phase to for all the teams to sit together and discuss the foundation of security and privacy issues and to analyze how to align quality and regulatory requirements with costs and business needs. Thus, even stakeholders can assess this.



1. **Establish security and privacy requirements:**

It is important to define and integrate security and privacy requirements as early as possible to make it easier to identify key milestones and deliverables to minimize disruptions to plans and schedules. Security and privacy analysis include assigning security experts, privacy criteria for an application and deploying a security work.

2. **Create quality bug bar:**

Creating quality bars i.e., minimum acceptable levels of security and privacy at the beginning helps team to understand the risk involved in this. Thus, they can apply the required standards right from the beginning.

3. **Perform security and privacy risk assessments:**

Examining software design, security features and privacy risks based on costs and regulatory requirements helps project to assess security before release.

Azure:

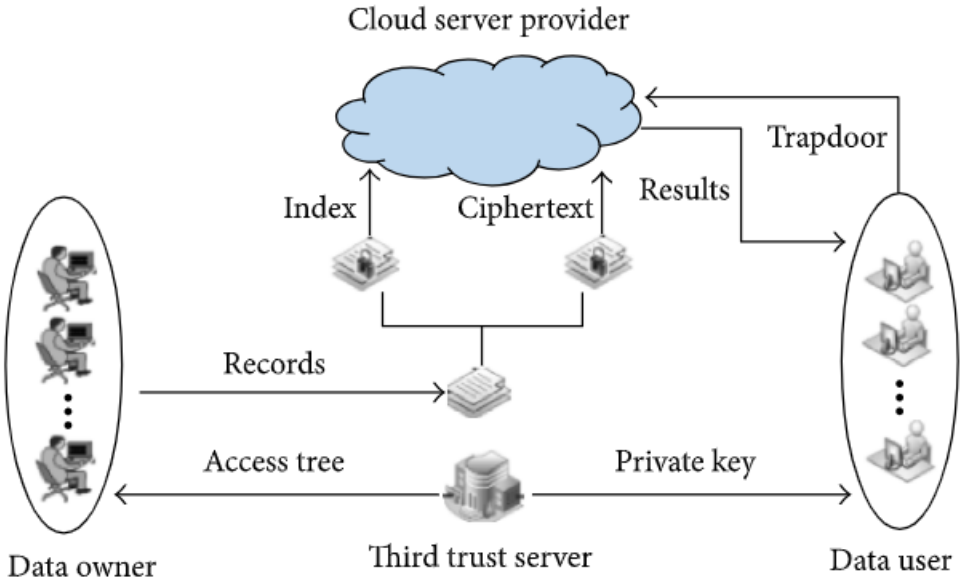
Azure Security:

Microsoft's Azure system provides a continuous support in detection and prevention of threats. It uses Multi-factor authentication for extra layer of defense. Even though, hackers accomplished their task in getting login-password, they still need other possessions like bio-metric or a phone. In addition to this, Microsoft Azure offers access control, patches, anti-malware,

monitoring, vulnerability scanning and configuration management.

There exists a solution accelerator which secures the devices when they are out in the field by using a unique identity key and the process is similar to that of secure boot. During the manufacturing of these devices, device ID is set. Even though the altering the device ID is difficult, it is important to introduce logical ID so that, even when there is change in physical parts, logical ID would remain same. Other security features include:

- Device would not connect to any network without authenticating. It will first check the connection with the IoT hub and once the connection is securely established, data from cloud to devices and its reciprocal happens.
- As device IDs are used for authentication and authorization, their credentials can be revoked instantly.



Azure Connection Security:

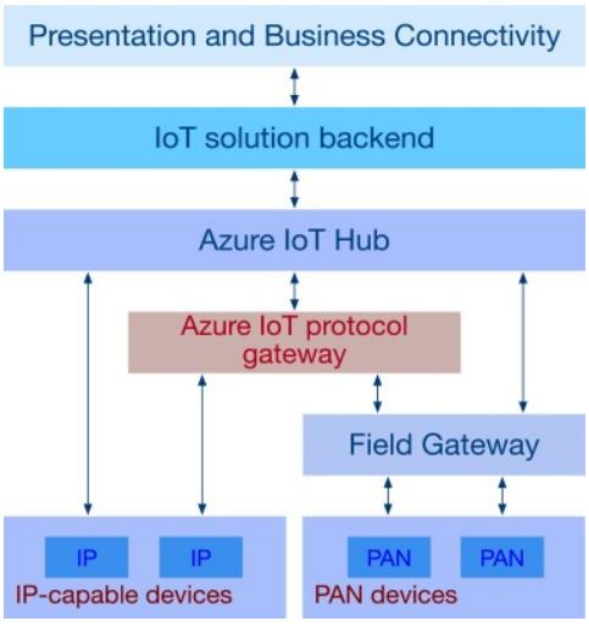
For any IoT device, its messaging endurance is the most prominent feature. As we know, how vulnerable devices that connect to the internet can be, this feature stands out in the security features. In addition to the above discussed authenticated messaging system, Azure maintains additional messaging durability by caching messages in the IoT Hub for about seven days for telemetry and two days for commands.

Azure even supports industry standard secure version of HTTP protocol, HTTPS, AMQP (Advanced Message Queuing protocol) and MQTT (Message Queuing Telemetry Transport). It enables secure connection of both IP-enabled and non-IP-enabled devices. IP-enabled devices can directly communicate with the cloud and thereby devices are being communicated. However, non-IP-devices can only communicate short distances like Bluetooth device, and thus a field gateway is used for protocol translation and thereby enabling secure bi-direction communication with the

cloud. This flow can be easily depicted using the following diagram:

Other connection security features include:

- The communication pathway between Azure's IoT hub and device is secured by TLS (Transport Layer Security) with hub authenticating it using, X.509 protocol.
- Azure's IoT hub do not open any connection with the device by itself, as this may promote other malware connections. Thus, only devices can initiate the connection.
- Azure IoT hub maintains queue for each device such that the commands are stored and ready to be sent to the device when ever it tries to make connection with the hub. These commands are stored for about two days, thus even problems like low power and connectivity can be see as a case when sending a command.



Security Architecture:

It is important to understand the potential threats at the time of system design as discussed in the life cycle above. In addition to this, understanding how a hacker thinks and what are the ways that he can hack into the system, would help us in coming up with a plan of action in developing the security features right from the design phase. This is called threat modeling. Its objective is to understand how a hacker might compromise the system and there by coming up with mitigation plan. Instead of coming up with security updates when the device is being used by the customer or altering features at the time of deployment, it is highly cost effective and efficient to consider threat modeling at the time of design phase.

Even though, threat modeling is a documentation of the discussions made, it plays a vital role at the time of continuity of knowledge and modeling the testing features. Most of the time, developers are much more concerned about capturing functional features at the time of design, this

identification of non-obvious way of attacking a device must equally be given importance.

The threat modeling process is of four steps:

- Model the application
- Enumerate Threats
- Mitigate threats
- Validate the mitigations

The four core elements of a threat model are:

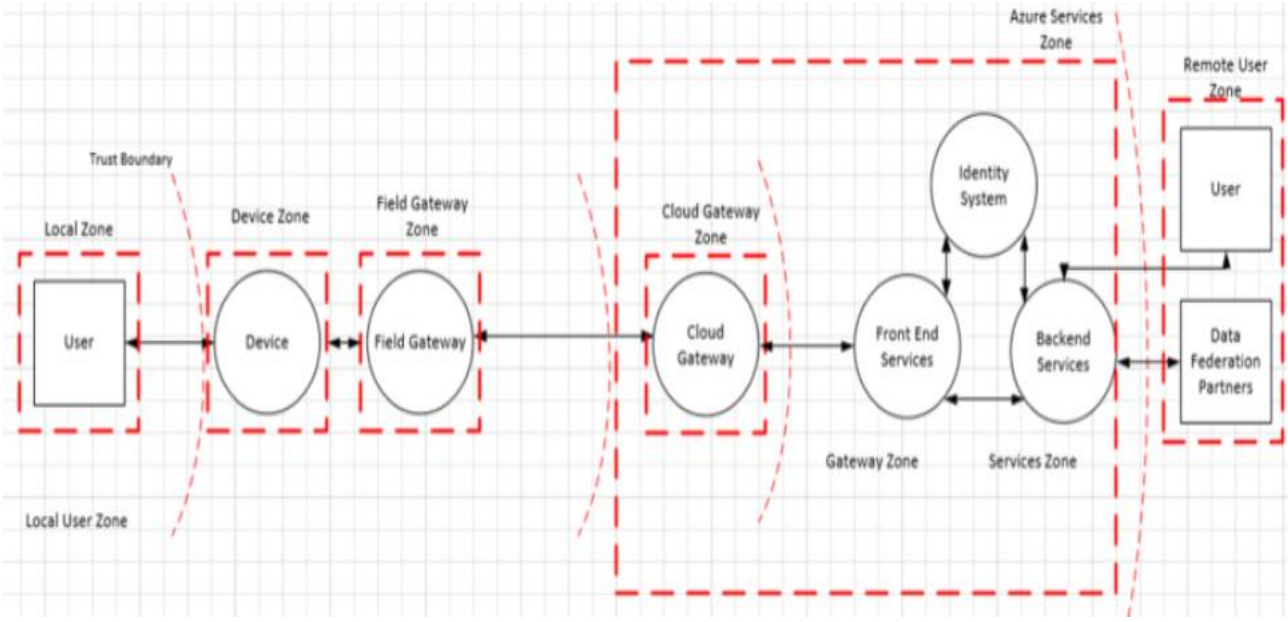
- Data flow (communication flow between devices and IoT hub)
- Data stores
- Gate ways and sensors
- External factors

Zones in IoT:

It is important to understand the potential interaction surface areas and interaction patterns to provide a framework for securing digital access into the devices. To make this happen, one should be clear with the difference between device control and device data. “Device control” is the information given to the device whose goal is to influence its behavior towards its environment. “Device data” is the information that a device emits about its state and environment.

To enhance the security for IoT devices, these devices were divided into zones or components. These zones include:

- Device
- Service
- Field gateway
- Cloud gateway



Each zone mentioned above has their own data, authentication and authentication requirements. These zones are used to trim down the level of damage occurred due to attack as these can be used to isolate damage and restrict the impact of low trust zones on higher trust zones.

In the above diagram, dotted lines indicate trust boundary which represents the transition of data from one source to another. During this transition, the information may be subjected to repudiation, information disclosure, denial of service, spoofing, tampering and elevation of privilege.

Device zone:

Device zone includes the immediate physical space around the device where the device can be accessed physically or can be accessed through local network. A local network in this context may be a distinct and insulated network that includes any short-range wireless technology that allows peer to peer communication.

Field gateway zone:

The field gateway includes gateway itself, it is a device or some general-purpose server computer software which acts as a device data processing hub, device control system and communication enabler. The field gateway as show in above figure has two distinct surface areas of which one faces the device that is connect to it and will represent the zone inside and the other faces external parties and is the edge of the zone.

Cloud gateway zone:

Cloud gateway enables the communication from and to the device or gateways from the public networks typically cloud services. Here, cloud is referring to data processing system that is not bound to the system that which is connected to the device hub.

Service zone:

As the name its self indicates that any software component that is interfacing with the device through cloud for data analysis and command control. They act like a mediator and are used in authenticating the end users.

STRIDE:

Spoofing(S):

An attacker may extract cryptographic key or image ID of the device which is at software or hardware level and thereby access the system with a virtual or different device which may be disguised as the device from which the key has been taken from. There might be a situation where, an assassin tries to kill his pray who is in a hospital who is no insulin pump. All he has to do is

override insulin pump's program and thus make a move.

Denial of Service(D):

A device may not function properly if there is any interfering with its radio frequency or cutting of its power supply, in this situation, the device may not be able to report or communicate back to the server. Attacker may even flood the device with lot of junk or my open multiple parallel connections with the device thereby make the device in-operable.

Tampering(T):

An attacker may rewrite or replace the software running in the device, which may disguise as an original one if the attacker succeeded in getting the cryptographic key or image ID. This may lead to unwanted operations from a device.

Information disclosure(I):

If the above-mentioned tampering was successful, then that device may leak potentially important data or information once it retrieves the key required. It may lead to alter the communication flow.

Elevation of privilege(E):

A device that is specified to some kind of work can be forced to do some other work through attacking. There might be a situation where a medical device that is programmed to open through half the valve, may be tricked to open all the way.

Ways to mitigate STRIDE:

- Assigning identity to the device and authenticating the device.
- Making sure to apply tamperproof mechanism to the device, this would make it impossible to extract cryptographic keys or another image ID.
- Having authorization scheme for the device.
- Authenticating and protecting field gateway against tampering and setting up access control mechanism.
- Strong pairing of the external entities to the device.
- Security on protocol levels and thus trying to encrypt the traffic.
- Encrypting the messages with digital signatures, image ID.

IoT Security Tokens:

IoT hub avoids sending tokens through internet and thus authenticates device using security tokens. These tokens are limited in time validity and scope. Generally, IoT SDKs automatically generates tokens without requiring any additionally configuration and sometimes requires users to generate and use security tokens directly. Such instances include use of AMQP or HTTPS

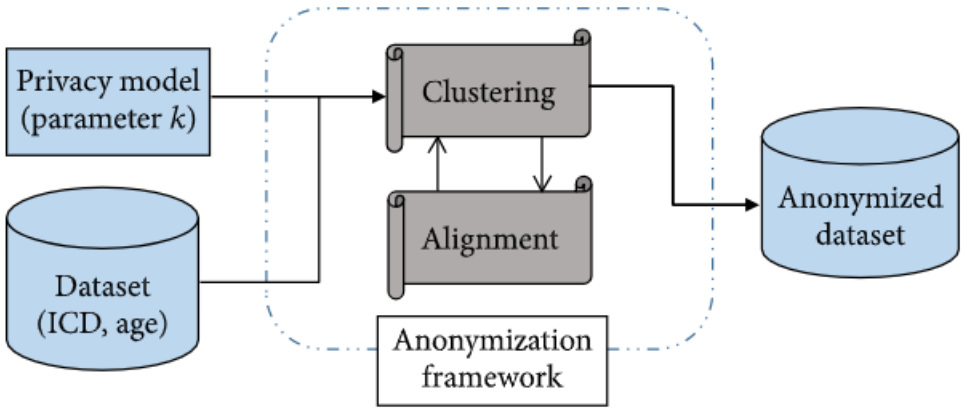
protocols. At the physical layer, an addition security layer device-based X.509 certificate with its associated public and private key pairs allows additional authentication. This certificate includes the information about the device. The connection between the device and the hub which is used for communication is protected using Transport Layer Security (TLS).

Data Anonymization:

Patient sensitive data can be divided into three categories:

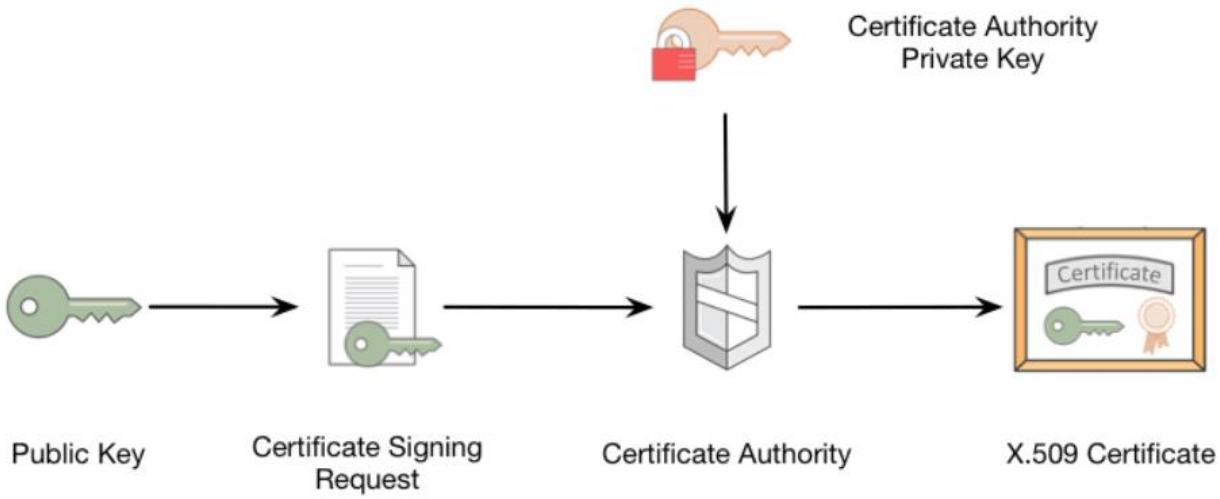
- Explicit Identifier
- Quasi Identifier
- Privacy Attributes

Explicit identifier uniquely identifies a patient with his information such as patient ID, cell number and other privacy attributes. Even with a combination of different quasi identifiers one can uniquely identify patient information. In the process of data publication, it is important that the sensitive attributes of the dataset are properly processed, to protect the patient information like patient health data, annual income, etc., Random perturbation technology and data anonymous technology are used to solve issues such as k-anonymity, l-diversity and confidence bounding. Many clustering algorithms can be used in this regard.

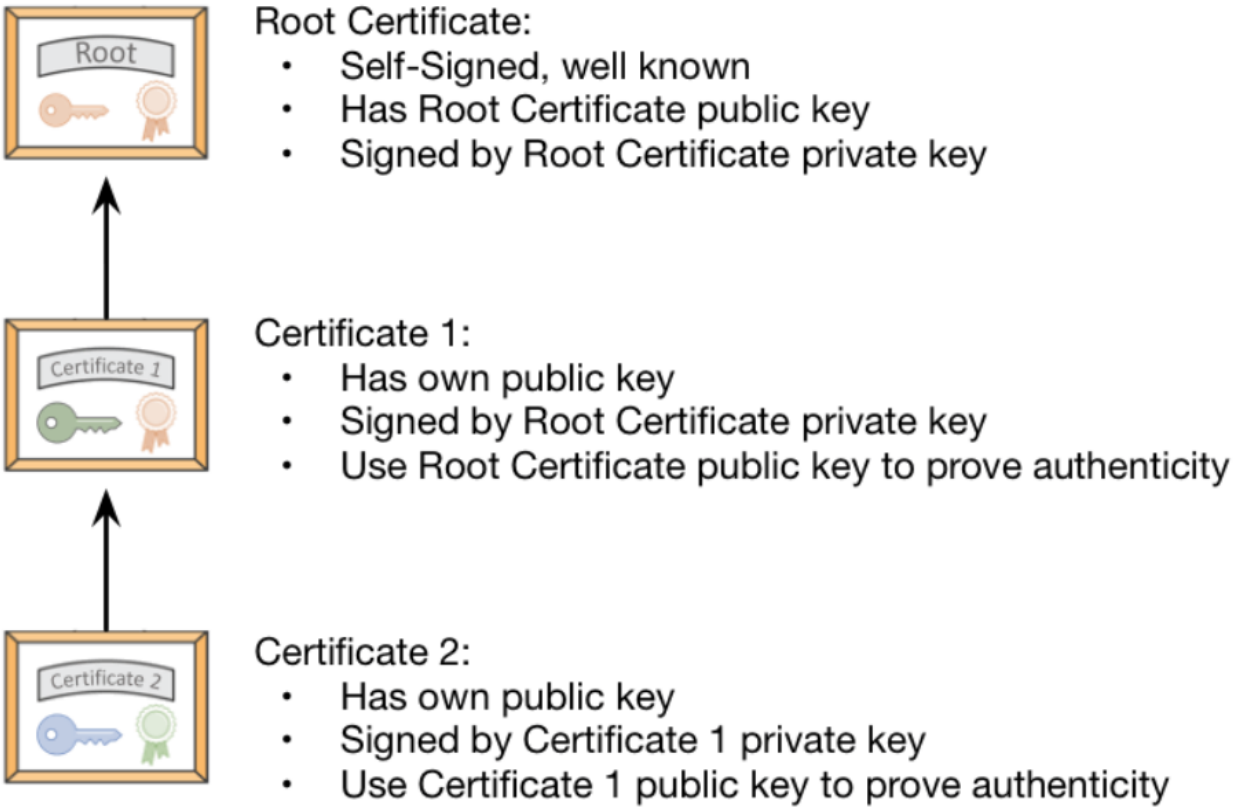


Amazon Web Service security model:

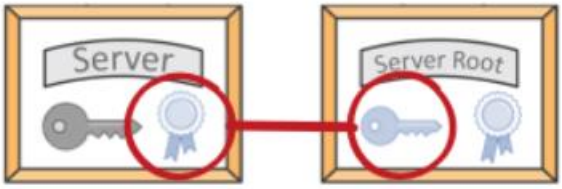
Even though the architecture is not much different from that of the Azure security model, in AWS, X.509 certificates are used in the communication. This certificate is the document that is used to prove the ownership of the public key. To know how X.509 certificate is created, lets first know what CSR and CA is. CSR is Certificate Signing Request which is sent by the user to CA i.e., Certificate Authority who is going to validate CSR which contains the information regarding the user and his public key. Once the identity is verified, CA creates a certificate with a private key signed on it.



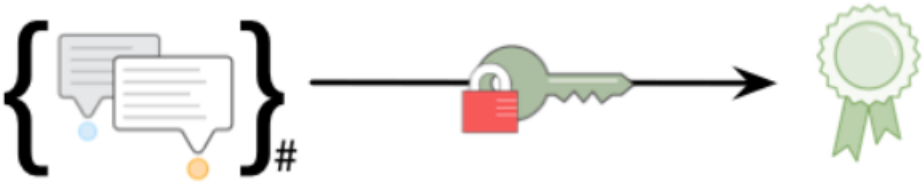
SO, how do we trust the key, if it is sent by CA is genuine. CA proves its ownership by signing its public key on X.509 certificate. CA certificate itself is signed by another CA and this cycle goes on till we find a root certificate as shown in the below figure



When the medical IoT device tries communicating the server, this starts with a TLS handshake, which will establish a secure channel between the device and the hub. When the hub receives the message, it will acknowledge back with a certificate. Now the IoT device cross checks the same with the help of root certificate that was installed in the device by the device manufacturer that cannot be altered.



After validating the key, the device trusts that the communication was from the AWS hub. Now, it authenticates itself with the AWS service and establishes a trusted channel for communication. To authenticate the channel that the device is being used, the device calculates the hash over those communication sessions. After this it calculates a digital signature for those hash using private key.



This signature is then sent to the AWS server. Now, AWS hub is in possession of the primary key of the device. It checks the accuracy of the digital signature using the public key.

Conclusion and recommendation:

For a company who is trying to provide a security solution to any medical device, it is important that, they work closely with the Original equipment manufacturers right from the device design phase. This is because, even though a device is provided with great security, it is waste of efforts, if the device hardware can easily be modeled and thus breached.

On working with Original equipment manufacturers, the above-mentioned security framework can be used to develop a strong security for any IoT device. Because it takes lot of efforts and costs more to work on security issues at the time of deployment and this may lead to come back to design phase. It is highly recommended that Original equipment manufacturers implement security analysis right at the device design phase.

Usage of secure communication channel protocols is the main essence of medical IoT device security and we can see platforms like Azure, AWS and Google Cloud coming up with different protocols, certification methods, image IDs and signature keys to secure the communication channels.

If we plan to choose services among Azure, AWS or Google Cloud as a platform to develop software for IoT device, the following table would help in clearing some doubts about firewall and identity management.

Security Services	AWS	Azure	Google
Authentication and Authorization	Identity and Access Management (IAM)	Active Directory Active Directory Premium	Cloud IAM Cloud Identity-Aware Proxy
Protection with Data Encryption	Key Management Service	Storage Service Encryption	-
Firewall	Web Application Firewall	Application Gateway	-
Identity Management	Cognito	Active Directory B2C	-
Cloud Services with Protection	Shield	DDoS Protection Service	-

In addition to this, it is advisable to consider the cost aspects. Even though Amazon Web Service (AWS) has about 62% of the market share, Azure and Google is showing a considerable growth with the time.

Public Cloud	Pricing	Models
Amazon Web Services	Per Hour – Rounded Up	On demand, Spot, and Reserved
Microsoft Azure	Per Minute – Rounded Up Commitments (Prepaid or Monthly)	On Demand- Short Term Commitments (Pre-paid or Monthly)
Google Cloud Platform	Per Minute – Rounded Up (Minimum 10 Minutes)	On Demand – Sustained Use

Reference:

- Understanding the AWS IoT Security model by Nick Corbett (Official blog): https://aws.amazon.com/blogs/iot/understanding-the-aws-iot-security-model/
- Cyber Security for Personal Medical Devices Internet of Things: https://ieeexplore-ieee-org.ezproxy.lib.usf.edu/document/6846193
- Security and privacy in the Medical Internet of Things: https://www.hindawi.com/journals/scn/2018/5978636/
- An access control management protocol for Internet of Things devices: https://www-sciencedirect-com.ezproxy.lib.usf.edu/science/article/pii/S1353485817300715
- Protecting IoT devices from cyberattacks: A critical missing piece by Icon Labs: http://www.smart2zero.com/news/protecting-iot-devices-cyberattacks-critical-missing-piece/page/0/9?sthash_mZ6JnrG9_mjjo=
- IoT platform for the health care industry (whitepaper): https://www.globallogic.com/gl_news/iot-platforms-for-the-health-care-industry/
- Securing cloud-connected devices with Cloud IoT and Microchip (vulnerabilities found by Google's Project Zero team): https://cloud.google.com/blog/products/gcp/securing-cloud-connected-devices-with-cloud-iot-and-microchip
- Protecting the internet of medical things: A situational crime-prevention approach by Murugan Anandarajan and Sarah Malik: https://www.cogentoa.com/article/10.1080/2331205X.2018.1513349
- Cloud Service Comparison by Neeru Jain: https://www.whizlabs.com/blog/aws-vs-azure-vs-google/
- Security of wearables fitness tracking IoT devices: https://ieeexplore-ieee-org.ezproxy.lib.usf.edu/document/6877073
- Assessing medical device vulnerabilities on the Internet of Things: https://ieeexplore-ieee-org.ezproxy.lib.usf.edu/document/8004903
- Internet of Medical Things: Dangers, Risks, and security problems by James Frew: https://www.makeuseof.com/tag/medical-internet-of-things-dangers/
- Evaluating your IoT Security by Microsoft: https://azure.microsoft.com/en-us/services/iot-hub/
- IoT Security Architecture by Robin Shahan and Bryan Lamos (Microsoft corporation): https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-architecture

- IoT Security Best Practices by Robin Shahan and Bryan Lamos (Microsoft corporation): https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-best-practices

- IoT Security from the ground up by Microsoft corporation: https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-ground-up

- Protecting privacy and security of patient health information: https://www.meditologyservices.com/blog-series-part-3-security-healthcares-space-junk-medical-device-iot-security