SECURITY AND PRIVACY IN THE INTERNET OF THINGS

IOT Security Threats

- There are three broad categories of threats:
  1. Capture
  2. Disrupt
  3. Manipulate.
- Capture threats → related to capturing the system or information.
- Disrupt threats →related to denying, destroying, and disrupting the system.
- Manipulate threats →related to manipulating the data, identity, time-series data, etc.

Passive Threats:

- The simplest type of passive threats in the IoT is that of eavesdropping or monitoring of transmissions with a goal to obtain information that is being transmitted.
- It is also referred to as capture attacks.
- Capture attacks are designed to gain control of physical or logical systems or to gain access to information or data items from these systems.

IOT SECURITY THREATS (CONTINUED)

- The ubiquity and physical distribution of the IoT objects and systems provide attackers with great opportunity to gain control of these systems.
- The distribution of smart objects, sensors, and systems results in self-advertisements, beacons, and mesh communications, providing attackers greater opportunity to intercept or intercede in information transmission within the environment.
- Moreover, the frequency of the data transmissions, data models, and formats help attackers in cryptanalysis.

IOT SECURITY THREATS (CONTINUED)

Active Threats :

- **Masquerading:** an entity pretends to be a different entity. This includes masquerading other objects, sensors, and users.
- **Man-in-the-middle:** when the attacker secretly relays and possibly alters the communication between two entities that believe that they are directly communicating with each other.
- **Replay attacks:** when an intruder sends some old (authentic) messages to the receiver.
- **Denial-of-Service (DoS) attacks**: when an entity fails to perform its proper function or acts in a way that prevents other entities from performing their proper functions.

IOT SECURITY REQUIREMENTS

- The basic security properties that need to be implemented in IoT are listed :

i. Confidentiality: Transmitted data can be read only by the communication endpoints;

ii. Availability: The communication endpoints can always be reached and cannot be made inaccessible;

iii. Integrity: Received data are not tampered with during transmission, and assured of the accuracy and completeness over its entire lifecycle;

iv. Authenticity: Data sender can always be verified and data receivers cannot be spoofed.

v. Authorization: Data can be accessed only by those allowed to do so and should be made unavailable to others.

Scale

- The important requirement is the scale in which an IoT environment is expected to grow.
- The population of entities is expected to grow exponentially as users embrace more smart and connected objects and devices, more sensors are deployed, and more objects are embedded with intelligence and information.
- Each entity, depending on its nature, characteristics, carries with it an associated set of protocols, channels, methods, data models, and data items, each of which is subject to potential threat.

SCALE (CONTINUED)

- This increased scale has the effect of expanding the target surface.
- As noted earlier, the scale and complexity at each level of the IoT model determine the amount of compute and storage requirements, and hence the cost and power budget.
- The trade-off between cost and resources determines the availability of resources for system security, cryptographic algorithms, key size, and methods.

IP PROTOCOL-BASED IOT

- The use of IP technologies in IoT brings a number of basic advantages such as:
  - A seamless and homogeneous protocol suite, and proven security architecture.
  - It also simplifies the mechanisms to develop and deploy innovative services by extending the tested IP-based frameworks.
  - It leads to a phenomenon called "expansion of attack surface."
  - It implies that when we connect the previously unconnected—by introducing new devices that stream context sensitive data, by placing data in mobile cloud, or by pushing computing to edge devices—new points of ingress for security threats inevitably materialize.

HETEROGENEOUS IOT

- Another important design consideration in the IoT is how the connected things can work together to create value and deliver innovative solutions and services.
- IoT can be a double-edged sword.
- Although it provides a potential solution to the innovation imperative, it can also significantly boost operational complexity if not properly integrated with key organizational processes.
- Security processes should also be properly designed to align with the organization processes.
- The complex operational technologies make it difficult for designing a robust security architecture in IoT.

HETEROGENEOUS IOT (CONTINUED)

- It is a common opinion that in the near future IP will be the base common network protocol for IoT.
- This does not imply that all objects will be able to run IP.
- In contrast, there will always be tiny devices, such as tiny sensors or Radio- Frequency Identification (RFID) tags, that will be organized in closed networks implementing very simple and application-specific communication protocols and that eventually will be connected to an external network through a proper gateway.
- In short, the heterogeneous characteristics of the networks make it harder to
  - implement certain IP-based security systems such as symmetric cryptosystems.

LIGHTWEIGHT SECURITY

These mechanisms rely on the following:

- Cryptographic ciphers such as Advanced Encryption Standard (AES), Secure Hash Algorithm (SHA2), and the public-key ciphers RSA and elliptic-curve cryptography (ECC).
- Transport Layer Security (TLS) protocol, and predecessor Secure Sockets Layer (SSL) protocol, which provide authentication and information encryption using the ciphers mentioned.
- Public-Key Infrastructure (PKI) provides the building blocks for authentication and trust through a digital certificate standard and Certificate Authorities (CA).

IOT Security Overview:

- *Necessary background on the IoT control protocols.*
- *It also discusses the key concepts on IoT security that includes identity management, authentication, authorization, privacy, trust, and governance for IoT networks.*
  - **IoT Protocols**
  - **Network And Transport Layer Challenges**
  - **IoT Gateways And Security**
  - **IoT Routing Attacks**

IOT SECURITY OVERVIEW

Table 10.2 Security Mechanisms to Mitigate the Threats in the IoT Networks

Threats/Security Mechanism	Data Privacy	Data Freshness	Source Authentication	Data Integrity	Intrusion Detection	Identity Protection
Capture						
Physical systems						X
Information	X			X		X
Disrupt						
DoS Attack		X	X		X	
Routing attack					X	
Manipulate						
Masquerading	X		X	X		X
Replay attack		X	X	X	X	
Man-in-the-middle			X	X	X	

Figure: The taxonomy of security attacks, threats, and security mechanisms.

IOT PROTOCOLS

- There are currently IETF (Internet Engineering Task Force) working groups focusing on extending existing protocols for resource-constrained networked environments.
- These are:
  - CoRE,
  - 6LoWPAN – (IPv6 over Low-power WPAN).
  - Routing Over Low power and Lossy networks (ROLL), and
  - Light-Weight Implementation Guidance (LWIG) working groups.
- Significant reasons for proper protocol optimizations and adaptations for resource-constrained objects are targeted toward protocol compression to fit into smaller Maximum Transmission Units (MTU), thereby reducing power consumption with smaller packets, elimination of fragmentation, and reducing the handshake messages.

IOT PROTOCOLS (CONTINUED)

Table 10.3 Bluetooth Smart Device Protocol Stack

Application layer	CoAP MQTT
Transport layer	UDP TCP
Network layer	IPv6 ICMPv6 RPL
Adaptation layer	Bluetooth Smart 6LoWPAN
Physical and link layer	IPSP

Network And Transport Layer Challenges

- The IPSec uses the concept of a Security Association (SA), defined as the set of algorithms and parameters (such as keys) used to encrypt and authenticate a particular flow in one direction.
- To establish a SA, IPSec can be preconfigured (specifying a preshared key, hash function, and encryption algorithm) or can be dynamically negotiated by the IPSec Internet Key Exchange (IKE) protocol.
- The IKE protocol uses asymmetric cryptography, which is computationally heavy for resource-constrained devices.

Network And Transport Layer Challenges (Continued.)

- To address this issue, IKE extensions using lighter algorithms should be used.
- Data Overhead is another problem for IPSec implementations in IoT environments.
- This is introduced by the Extra Header Encapsulation of IPSec AH and/or Encapsulating Security Payload (ESP), and can be mitigated by using header compression.
- CoAP proposes to use the D T L S protocol to provide end-to-end security in IoT systems. (DTLS → Datagram Transport Layer Security , A communication protocol)
- The DTLS protocol provides a security service similar to TLS, but on top of UDP.
- This is highly suitable for IoT environments due to its usage of UDP as transport protocol.

Network And Transport Layer Challenges (Continued.)

- This results in avoidance of problems from the use of TCP in network-constrained scenarios that are caused due to the extremely variable transmission delay and loss links.
- DTLS is a heavyweight protocol and its headers are too long to fit in a single IEEE 802.15.4 MTU.
- 6LoWPAN provides header compression mechanisms to reduce the size of upper layer headers.
- 6LoWPAN header compression mechanisms can be used to compress the security headers as well.

IOT GATEWAYS AND SECURITY

- Connectivity is one of the important challenges in designing the IoT network.
- The diversity of end points makes it very difficult to provide IP connectivity.
- It is important that non-IP devices too have a mechanism to connect with IoT.
- The IoT gateways can simplify IoT device design by supporting the different ways nodes natively connect, whether this is a varying voltage from a raw sensor, a stream of data over an inner integrated circuit (I2C) from an encoder, or periodic updates from an appliance via Bluetooth.
- *(Gateway → a network node, that forms a passage between two networks operating with different transmission protocols.)*

IOT GATEWAYS AND SECURITY (CONTINUED)

- Gateways effectively mitigate the great variety and diversity of devices by consolidating data from disparate sources and interfaces and bridging them to the Internet.
- The result is that individual nodes do not need to bear the complexity or cost of a high-speed Internet interface in order to be connected.
- There are several ways that an IoT gateway can extend connectivity to nodes as described.

IOT GATEWAYS AND SECURITY (CONTINUED)

i. The network nodes connect to the IoT via a gateway.

- The nodes themselves are not IP-based and thus cannot directly connect to the Internet/WAN.
- Rather, they use either wired or wireless PAN technology to connect to the gateway with a less expensive and less complex mode of connectivity.
  - *(Wireless PAN – Wireless Personal Area Network (WPAN))*
- The gateway maintains an IoT agent for each node that manages all data to and from nodes.
- In this case, application intelligence can also be located in the gateway.

IOT GATEWAYS AND SECURITY (CONTINUED)

ii. The nodes can also connect directly to the Internet using a WAN connection such as Wi-Fi or Ethernet.

- **The gateway serves primarily as a router; in fact, it can be simply a router when nodes have their own IoT agent and autonomously manage themselves.**

- **Alternatively, the nodes can connect directly to the Internet using a PAN connection such as 6LoWPAN.**
  - **In this case, the gateway serves as a translation point between the PAN and WAN.**

IOT GATEWAYS AND SECURITY (CONTINUED)

- Many IoT applications handle potentially sensitive data.
- Data collected from location services, for example, need to be protected from hacking.
- Similarly, medical devices need to maintain the privacy of individuals.
- In the context of the IoT gateway architecture, the security processing and mechanisms can be offloaded from nodes to the gateway to ensure proper authentication, protecting exchanges of data, and safeguarding intellectual property.
- This enables IoT nodes to implement greater security than could be economically implemented in individual end points.

IoT Routing Attacks

- **Threats** arising due to the physical nature of IoT devices can be mitigated by appropriate physical security safeguards, whereas **secure communication protocols** and **cryptographic algorithms** are the only way of coping with the fact that they arise due to IoT devices communicating with each other and the external world.
- For the later, **IoT devices** can either **run** the standard **TCP/IP protocol** stack, if their computational and power resources allow, or can run adaptions which are optimized for lower computational and power consumption.

IoT Routing Attacks (Continued)

- Some well known Routing Attacks On IoT are as follows:

1) Selective-forwarding attacks
2) Sinkhole attacks
3) Hello flood attacks
4) Wormhole attacks
5) Clone Id and Sybil attacks

IoT Routing Attacks (Continued)

1. selective-forwarding attacks

- With **selective-forwarding attacks**, it is possible to launch DoS attacks where malicious nodes selectively forward packets.
- This attack is primarily targeted to disrupt routing paths.
- Eg:- an attacker could forward all RPL control messages and drop the rest of the traffic.
- This attack has severer consequences when coupled with other attacks such as sinkhole attacks.
- One of the solutions to guard against selective forwarding attacks is to create disjoint paths between the source and the destination nodes.
- Another effective counter measure against selective-forwarding attacks is to make sure the attacker cannot distinguish between different types of traffic, thus forcing the attacker to either forward all traffic or none.

IoT Routing Attacks (Continued)

2. Sinkhole attacks:

- a malicious node advertises a fraudulent routing path with a seemingly favorable route metric and attracts many nearby nodes to route traffic through it.
- An intrusion detection system could be hosted in the 6 L B R and can utilize information from multiple DODAGs to detect sinkhole attacks.

3. hello-flood attack:

- In tthis attack, The HELLO message refers to the initial message a node sends when joining a network.
- By broadcasting a HELLO message with strong signal power and a favorable routing metric, an attacker can introduce himself as a neighbor to many nodes, possibly the entire network.
- A simple solution to this attack is for each HELLO message the link is checked to be bidirectional.

IOT ROUTING ATTACKS (CONTINUED)

- 4. **wormhole**
- A **wormhole** is an out-of-band connection between two nodes using wired or wireless links.
- *A severe network layer threat where **malicious nodes create tunnels** to **replay wireless information,** compromising network integrity.*
- Wormholes can be used to forward packets faster than via normal paths.
- A wormhole created by an attacker and combined with another attacks, such as sinkhole, is a serious security threat.

i. One approach is to use separate link-layer keys for different segments of the network.

- This can counteract the wormhole attack, as no communication will be possible between nodes in two separate segments.

i. Also, by binding geographic information to the neighborhoods it is possible to overcome a wormhole.

IOT ROUTING ATTACKS (CONTINUED)

- In a **clone-ID attack,** an attacker copies the identities of a valid node onto another physical node.
- This can, for example, be used in order to gain access to a larger part of the network or in order to overcome voting schemes.
- In a **Sybil attack,** which is similar to a clone ID attack, an attacker uses several logical entities on the same physical node.
- Sybil attacks can be used to take control over large parts of a network without deploying physical nodes.
- By keeping track of the number of instances of each identity it is possible to detect cloned identities.
- It would also be possible to detect cloned identities by knowing the geographical location of the nodes, as no identity should be able to be at several places at the same time.

BOOTSTRAPPING AND AUTHENTICATION

- Bootstrapping and authentication controls the network entry of nodes.
- Authentication is highly relevant to IoT and is likely to be the first operation carried out by a node when it joins a new network, for instance, after mobility.
- It is performed with a (generally remote) authentication server using a network access protocol such as the PANA.
- For greater interoperability, the use of the EAP is envisioned.

BOOTSTRAPPING AND AUTHENTICATION (CONTINUED)

- Upon successful authentication, higher layer security associations could also be established (such as IKE followed by IPSec) and launched between the newly authenticated endpoint and the access control agent in the associated network.
- The Internet Key Exchange (IKEv2)/IPSec and the HIP reside at or above the
  - network layer.
- Both protocols are able to perform an authenticated key exchange and set up the IPSec transforms for secure payload delivery.
- Currently, there are also ongoing efforts to create a HIP variant called Diet HIP that takes loss low-power networks into account at the authentication and key exchange level.

AUTHORIZATION MECHANISMS

- The present day services that run over the Internet, such as popular social media applications, have faced and handled privacy-related problems when dealing with personal and protected data that might be made accessible to third parties.
- In the future, the IoT applications will face similar issues, and others that may be unique to the domain.
- The OAuth (Open Authorization) protocol has been defined to solve the problem of allowing authorized third parties to access personal user data.
- OAuth2.0 is an authorization framework that allows a third party to access a resource owned by a resource owner without giving unencrypted credentials to the third party.

AUTHORIZATION MECHANISMS (CONTINUED)

- For example, assume that a healthcare sensor or mobile app wants to access a Facebook profile to post status updates.
- There is no need to provide the Facebook credentials to the app; instead, the user logs into Facebook, and as a result the app is authorized to use Facebook on the user's behalf.
- The user can also revoke this authorization any time by deleting the privilege in the Facebook settings.
- The OAuth 2.0 protocol defines the following four roles.

AUTHORIZATION MECHANISMS (CONTINUED)

1. Resource Owner

- It is an entity capable of granting access to a protected resource.
- When the resource owner is a person, it is referred to as an end user.
- In the above example, this could be the end user of the healthcare device.

AUTHORIZATION MECHANISMS (CONTINUED)

2. Resource Server (Service Provider, SP)

- It is the server hosting the protected resources, capable of accepting and responding to protected resource requests using access tokens.
- In the example, this is the Facebook server.

AUTHORIZATION MECHANISMS (CONTINUED)

3. Client (Service Consumer, SC)

- It is the application making protected resource requests on behalf of the resource owner and with its authorization.
- The term client does not imply any particular implementation characteristics (eg, whether the application executes on a server, a desktop, or other devices).
- In this case, it is the healthcare sensor or mobile application.

AUTHORIZATION MECHANISMS (CONTINUED)

4. Authorization Server

- It is the server issuing access tokens to the client after successfully authenticating the resource owner and obtaining authorization.
- In this example, it would be the Facebook authorization server.

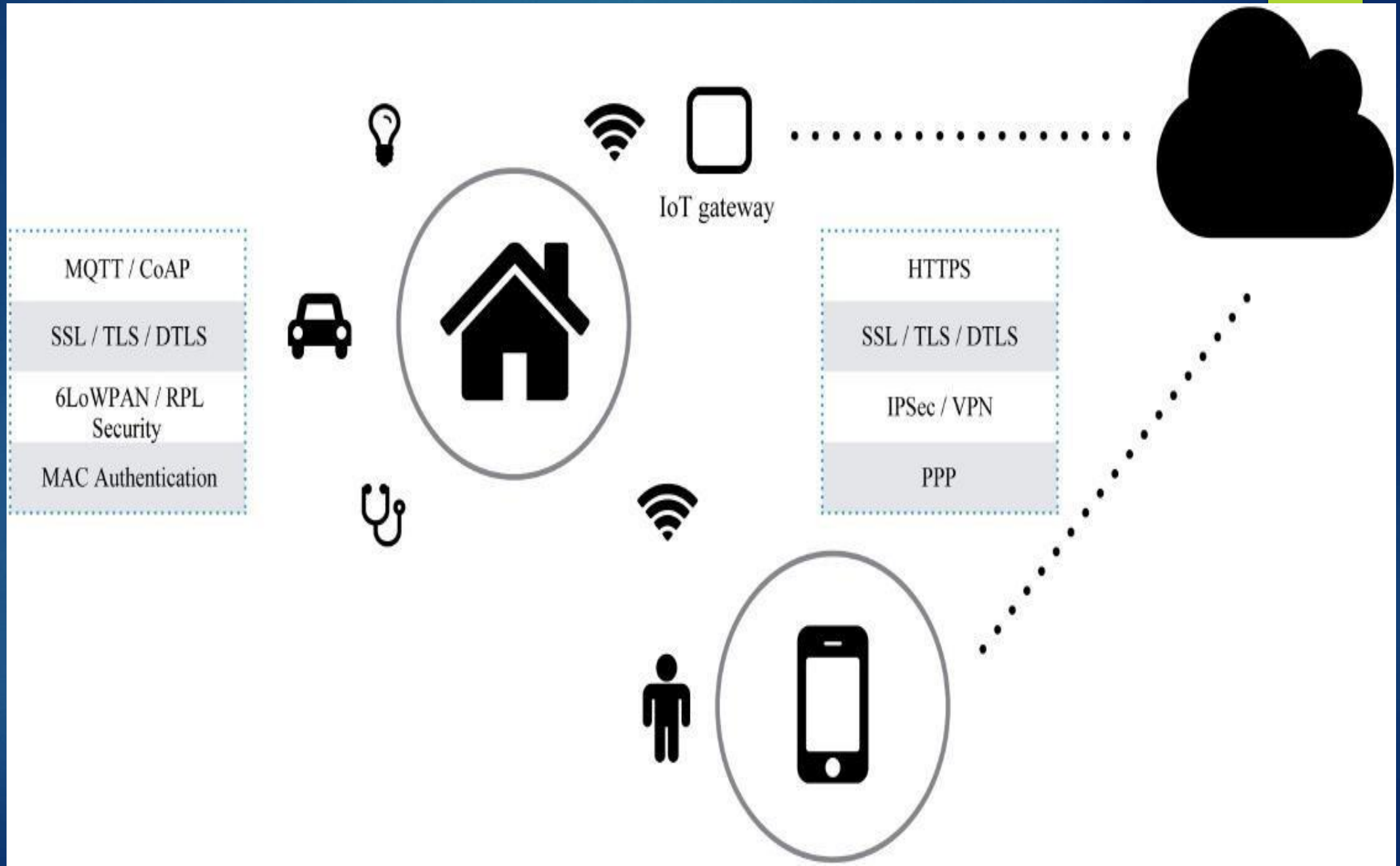
IOT OAS

- Note that the IoT devices may have challenges in implementation of OAuth due to the CPU intensive nature of cryptographic computations.
- A modified architecture called IoTOAS is proposed.
- In this approach, authorization-related functions are delegated to an external IoT-OAS authorization service, in order to minimize the memory and CPU requirements on the IoT device itself.
- An incoming OAuth secured request is forwarded to an IoT-OAS service for verification of the access token contained in the request.

IOT OAS (CONTINUED)

- The IoT-OAS service computes the digital signature of the incoming request using the appropriate scheme (PLAINTEXT/HMAC/RSA) and matches it with its internal store to verify the user and client credentials and permissions for resource access.
- It then provides an appropriate response back allowing or denying the requested access from the client.
- This approach enables the IoT device to focus on its own service logic and frees up computational resources from being overwhelmed by security and cryptographic implementations.
- The security protocols at each layer between different networks are shown in following diagram.

AN OVERVIEW OF IOT AND IP SECURITY PROTOCOLS



Security Frameworks For IoT

- In this section, we discuss some of the specific frameworks used for realizing a secure IoT system.
- The low capabilities of IoT devices in terms of their energy and computing capabilities, wireless nature, and physical vulnerability are discussed to be the contributing factors to some unique security vulnerabilities.
- In particular, we cover the tight resource constraints, protocol translation such as HTTP↔ CoAP, and end-to-end Security.
- Other important topics include the Architecture Framework Aspects: Distributed vs Centralized approach, bootstrapping identity and key interchange, privacy aware identification, mobility, and IP network dynamics.

Light Weight Cryptography

- The term lightweight cryptography refers to a family of cryptographic algorithms with smaller footprint, low energy consumption, and low computational power needs.
- Every designer of lightweight cryptography must cope with the trade-offs between security, cost, and performance.
- It is generally easy to optimize any two of the three design goals:
  - Security and cost, security and performance, or cost and performance;
- however, it is very difficult to optimize all three design goals at once.

Light Weight Cryptography (CONTINUED)

- When we compare lightweight cryptographic implementations, we can make a distinction between symmetric and asymmetric ciphers.
- Symmetric ciphers serve mainly for message integrity checks, entity authentication, and encryption.
- Asymmetric ciphers are computationally far more demanding, in both hardware and software. Asymmetric ciphers additionally provide key-management advantages and nonrepudiation.
- The performance gap on constrained devices such as 8-bit microcontrollers is huge.

Light Weight Cryptography (Continued)

- For example, an optimized asymmetric algorithm such as ECC performs 100– 1000 times more slowly than a standard symmetric cipher such as the AES algorithm, which correlates with a two to three orders of-magnitude higher power consumption.
- Symmetric-key cryptographic algorithms use the same key for encryption of a plain text and decryption of a message.
- The encryption key represents a shared secret between the parties that are involved in the secure communication.

ASYMMETRIC LWC ALGORITHMS

- Public-key (asymmetric) cryptography requires the use of a public-key and a private key.
- Public keys can be associated with the identity of a node by including them into a public certificate, signed by a Certification Authority (CA) that can be requested to verify the certificate.
- Public-key cryptography requires the significant effort of deploying a PKI.
- Moreover, asymmetric cryptography requires higher processing and long keys (at least 1024 bits for RSA) to be used.
- Alternative public-key cryptographic schemes, such as ECC, might require shorter keys to be used in order to achieve the same security than RSA keys.

ASYMMETRIC LWC ALGORITHMS (CONTINUED)

- However, because of these reasons, symmetric cryptography is preferred in terms of processing speed, computational effort, and size of transmitted messages.
- Public key can be used to setup symmetric keys to be used in subsequent communications.
- Lightweight cryptography algorithms are suitable for environments that do not have stringent security requirements and where the constraints on available hardware and power budget cannot be relaxed.

PRIVACY IN IOT NETWORKS

- The smart, connected objects will interact with both humans and other smart objects by providing, processing, and delivering all sorts of information and signals.
- All of these objects and their communications with the environment carry with them a risk to privacy and information leakage.
- Healthcare applications represent the most outstanding application of IoT.
- The lack of confidence regarding privacy results in decreased adoption among users and is therefore one of the driving factors in the success of IoT.

PRIVACY IN IOT NETWORKS (CONTINUED)

- The ubiquitous adoption of the wireless medium for exchanging data may pose new issue in terms of privacy violation.
- In fact, wireless channel increases the risk of violation due to the remote access capabilities, which potentially expose the system to eavesdropping and masking attacks.
- IoT devices and applications add a layer of complexity over the generic issue of privacy over the Internet, for example due to generation of traceable characteristics and attributes of individuals.
- IoT devices in healthcare present a major concern, since these devices and applications typically generate large volumes of data on individual patients through continuous monitoring of vital parameters.

PRIVACY IN IOT NETWORKS (CONTINUED)

- In this case, it is crucial to delink the identities of the device from that of the individual, through mechanisms such as data anonymization.
- Data anonymization is the process of either encrypting or removing personally identifiable information from data sets, so that the originator of the data remains anonymous.
- Similar to the preceding discussion of the OAuth protocol, digital shadows enable the individual's objects to act on their behalf, storing just a virtual identity that contains information about their parameters.
- Identity management in IoT may offer new opportunities to increase security by combining diverse authentication methods for humans and machines.
- For example, bio-identification combined with an object within the personal network could be used to open a door.

SECUREDATA AGGREGATION

- Homomorphic encryption is a form of encryption that allows specific types of computations to be executed on cipher texts and obtain an encrypted result that is the cipher text of the result of operations performed on the plain text.
- Applying the standard encryption methods presents a dilemma: If the data is stored unencrypted, it can reveal sensitive information to the storage/database service provider.
- On the other hand, if it is encrypted, it is impossible for the provider to operate on it.
- If data are encrypted, then answering even a simple counting query (for example, the number of records or files that contain a certain keyword) would typically require downloading and decrypting the entire database content.

SECURE DATA AGGREGATION (CONTINUED)

- A homomorphic encryption allows a user to manipulate without needing to decrypt it first.
- An example of homomorphic encryption is the RSA algorithm.
- Other examples of homomorphic encryption schemes are the ECC encryption, the ElGamal cryptosystem, and the Pailler cryptosystem.
- Homomorphic encryption has a lot of relevance to IoT networks, since privacy can be preserved at all stages of the communication, especially without the need for intermediate nodes to decrypt the information.
- For example, a lot of processing and storage can be eliminated at intermediate nodes by data aggregation with operations such as sums and averages.

SECURE DATA AGGREGATION (CONTINUED)

- This in turn results in lower power consumption, which is relevant for constrained environments.
- However, note that this type of homomorphic cryptosystems is more compute-intensive and needs longer keys to achieve a comparable security level than typical symmetric-key algorithms.

SECURE DATA AGGREGATION (CONTINUED)

Typically, secure data aggregation mechanisms require nodes to perform the following operations:

- at the transmitting node, prior to transmission, data are encrypted with some cryptographic function E
- at the receiving node, all received data packets are decrypted with the inverse cryptographic function $D = E^{-1}$ to retrieve the original data;
- data are aggregated with an aggregation function;
- prior to retransmission, aggregated data are encrypted through E and relayed to the next hop.

ENIGMA

- MIT Researchers, Guy Zyskind and Oz Nathan, have recently announced a project dubbed Enigma that makes a major conceptual step toward this Holy Grail of a fully homomorphic encryption protocol.
- A peer-to-peer network, enabling different parties to jointly store and run computations on data while keeping the data completely private is proposed.
- Enigma's computational model is based on a highly optimized version of secure multiparty computation, guaranteed by a verifiable secret-sharing scheme.
- For storage, it uses a modified distributed hash table for holding secret-shared data.

ENIGMA (CONTINUED)

- An external block chain is utilized as the controller of the network, manages access control, identities, and serves as a tamper-proof log of events.
- Security deposits and fees incentivize operation, correctness, and fairness of the system.
- Similar to Bitcoin, Enigma removes the need for a trusted third party, enabling autonomous control of personal data.
- For the first time, users are able to share their data with cryptographic guarantees regarding their privacy.

ENIGMA (CONTINUED)

- The typical use case of Enigma would be for interactions between hospitals and health-care providers who store encrypted patient data as per HIPAA regulations.
- Research organizations and pharmaceutical companies would benefit from access to these data for clinical analysis.
- For example, a hospital can encrypt its data and store it in the cloud, where potentially other universities, pharma companies, and insurance companies could access it with permission from the originating hospital.
- With the usage of Enigma, note that there is no need for the originating hospital to first decrypt and anonymize the data, it only needs to authorize the third party for access.

ZERO KNOWLEDGE PROTOCOLS

- Zero-knowledge protocols allow identification, key exchange and other basic cryptographic operations to be implemented without leaking any secret information during the conversation and with smaller computational requirements than using comparable public-key protocols.
- Thus Zero-knowledge protocols seem very attractive especially in the context of IoT networks, especially for some applications like smart cards.
- Zero-knowledge protocols have been claimed to have lighter computational requirements than, for example, public-key protocols.

ZERO KNOWLEDGE PROTOCOLS (CONTINUED)

- The usual claim is that zero-knowledge protocols can achieve the same results than public-key protocols with one to two orders of magnitude less (1/10, 1/100) computing power.
- A typical implementation might require 20–30 modular multiplications (with full-length bit strings) that can be optimized to 10–20 with precalculation.
- This is much faster than RSA.
- The memory requirements seem to be about equal: to have very high security with zero-knowledge protocols, we need very long keys and numbers, so in memory terms, the requirements may not be very different.

PRIVACY IN BEACONS

- Beacon in wireless technology is the concept of broadcasting small pieces of information.
- The information may be anything, ranging from ambient data to vital signs such as body temperature, blood pressure, pulse, and breathing rate or microlocation data such as asset tracking.
- Based on the context, the transmitted data maybe static or dynamic and change over time.
- The Bluetooth beacon opens a new world of possibilities for location awareness, and countless opportunities for smart applications.
- Beacons are becoming one of the key enablers of the IoT.

PRIVACY IN BEACONS (CONTINUED)

- One kind of beacon is a low energy Bluetooth transmitter or receiver.
- The power efficiency of Bluetooth Smart makes it perfect for devices needing to run off a tiny battery for long periods.
- The advantage of Bluetooth Smart is its compatibility to work with an application on the smartphone or tablet you already own.
- An important use case of beacons is to obtain context-specific observations and repeated measurements over time.
- Most data collected from beacons are correlated in time, which might cause serious threats to data security and user privacy.

PRIVACY IN BEACONS (CONTINUED)

- Security and privacy issues specific to beacons and time series data transmitted from them are emerging areas of research interest.
- There are both advantages and disadvantages of security based on the difficulty of an underlying computation problem and information theoretic security, which is based on lack of information content.
- A more basic measure of the information-theoretic security is the inherent information available for exploitation by an adversary, independent of how the adversary exploits it or indeed any assumed computational limitations of the adversary.

PRIVACY IN BEACONS (CONTINUED)

- A new measure of information theoretic measure such as conditional entropy is shown to be suited for evaluating the privacy of perturbed real-world time-series data, compared with other existing measures.
- Much of the research and study of privacy issues in ubiquitous computing systems is applicable to the IoT.
- Establishing meaningful identity, using trusted communication paths, and protecting contextual information is all very important to ensure the protection of user privacy in this environment.
- Anonymous communication techniques and the use of pseudonyms to protect user privacy while also working on metrics to assess user anonymity.

PRIVACY IN BEACONS (CONTINUED)

- A novel approach by hiding identity from the applications that utilize it in order to better protect the user consuming those services.
- New technologies that enable the bootstrapping of trust, and subsequently, the calculation of trust metrics that are better suited to mobile, ad-hoc networks is proposed.
- The model showcases the inherent problems with establishing trust in ad-hoc networks like those in the IoT where new sensors, services, and users are constantly introduced and asked to share data.

PRIVACY IN BEACONS (CONTINUED)

- Finally, applications in the IoT, which will be enabled by a ubiquitous computing and communications infrastructure, will provide unobtrusive access to important contextual information as it pertains to users and their environment.
- Clearly, the successful deployment of such applications will depend on our ability to secure them and the contextual data that they share.

PRIVACY IN BEACONS (CONTINUED)

- One example of sensitive contextual information is location.
- When location-aware systems track users automatically, an enormous amount of potentially sensitive information is generated and made available.
- Privacy of location information is about both controlling access to the information and providing the appropriate level of granularity to individual requestors.
- The Location Services Handbook explores a variety of location-sensing technologies for cellular networks and the coverage quality and privacy protections that come with each.