Internet of Things (IoT)

Security in IoT

Security in IoT Survey: https://ieeexplore.ieee.org/document/7005393

Dr. Manas Khatua

Associate Professor

Dept. of CSE, IIT Guwahati

E-mail: manaskhatua@iitg.ac.in

"Peace comes from within. Do not seek it without." – **Gautama Buddha**

Introduction

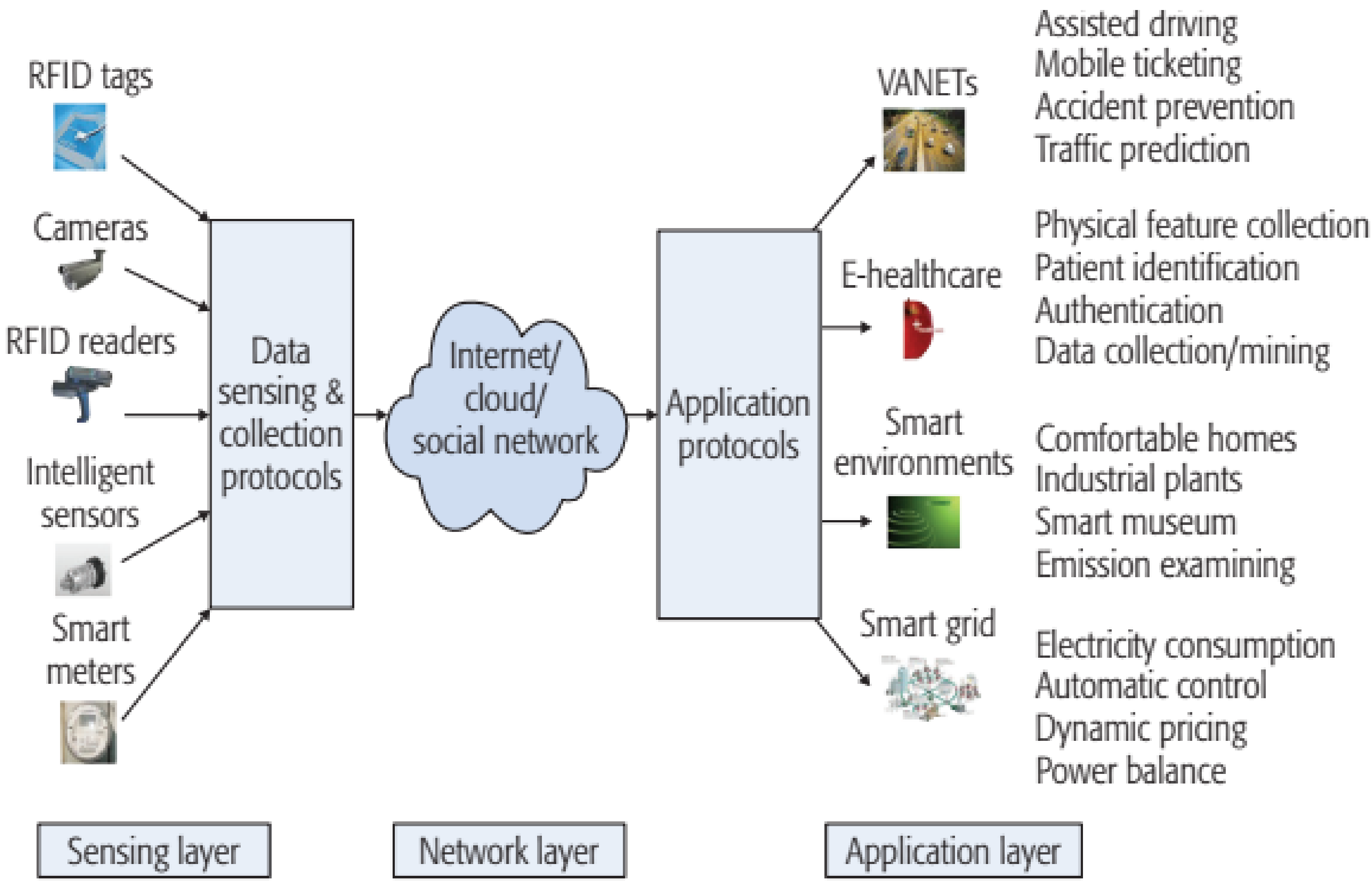
- IoT
  - uses low-power low-rate wireless communication protocol stack
  - millions of sensing and actuating devices are attached

- IoT security is the act of securing IoT devices and the networks they're connected to.

- Security solutions employed in TCP/IP are ill suited for IoT
  - Because of the constrained IoT node and network

Cont…

Traditional Network architecture of cloud-based IoT Solutions.



Source: Zhou et. al., "Security and Privacy for Cloud-Based IoT: Challenges, Countermeasures, and Future Directions" IEEE Comm. Magazine, Jan. 2017, pp. 26-33.

Security Requirements

Fundamental Requirements:

- Confidentiality
  - refers to protecting information from being accessed by unauthorized parties
- Integrity
  - Data must not be changed in transit.
- Availability
  - is a guarantee of reliable access to the information by authorized people.
- Authentication
  - merely ensures that the individual is who he or she claims to be
- Non-repudiation
  - is the assurance that someone cannot deny the validity of something
- Resilience
  - is the ability to prepare for, respond to and recover from an attack.

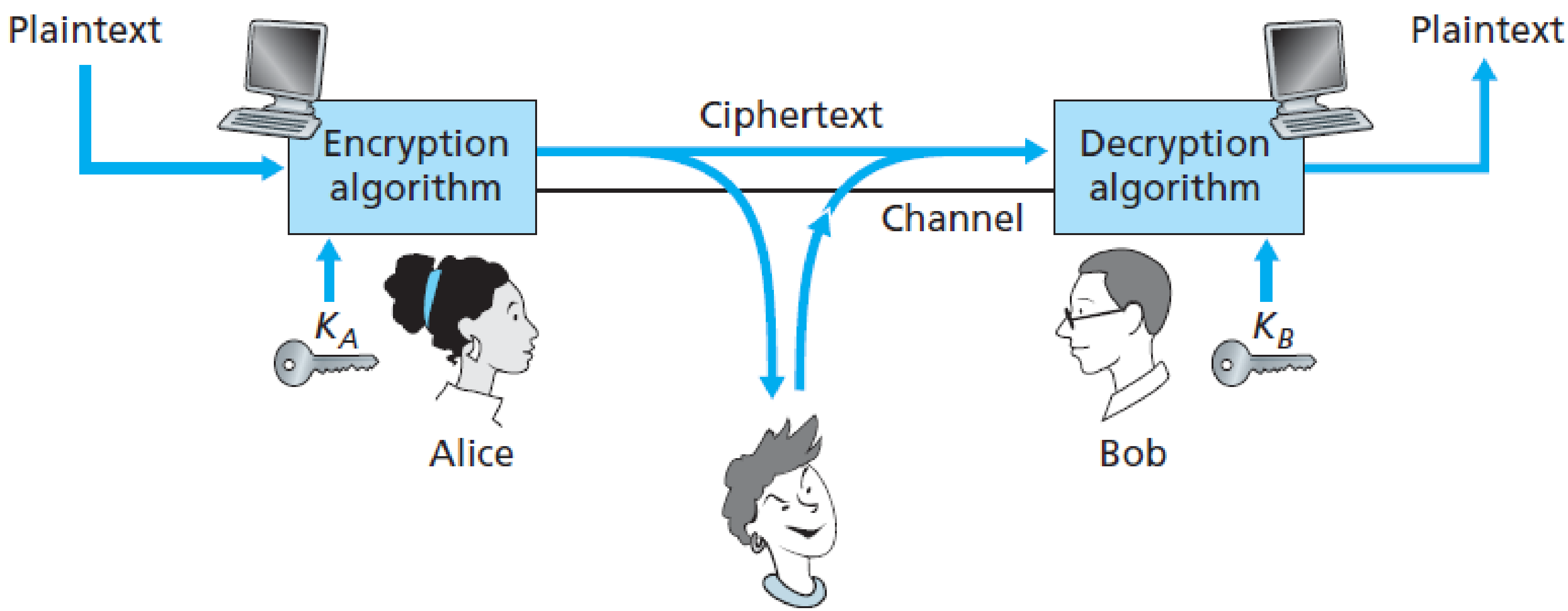
Few more Requirements - Privacy ; Anonymity; Accountability; Trust

Common Security Attacks

- *Snooping :* unauthorized access or interception of data.
- *Traffic Analysis :* obtain some other types of information by monitoring online traffic.
- *Modification :* modifies the information to make it beneficial to the attacker
- *Masquerading* or *spoofing* : the attacker impersonates somebody else.
- *Replaying :* the attacker replays the obtained/snooped copy of a message later on.
- *Repudiation :* The sender of the message might later deny that it has sent the message OR the receiver of the message might later deny that it has received the message.
- *Denial of Service* (DoS)*:* attacker may slow down or totally make unavailable the service of a legitimate system.

Cryptography

- It has a long history dating back at least as far as Julius Caesar (dated: 100 BC).
- Cryptographic techniques allow a sender to disguise data so that an intruder can gain no information from the intercepted data.



cont…

- Broadly classified into two categories: Symmetric and Asymmetric Key Cryptography

- Symmetric Key Cryptography:
  - ✓ It allows both the sender and receiver to utilize the secret but same key for both the encryption and decryption processes.

- Asymmetric Key Cryptography:
  - ✓ It involves the use of two distinct keys, namely the Public Key and Private Key. Public Key is know to both, but the Private Key is known to either sender or receiver, but not both.

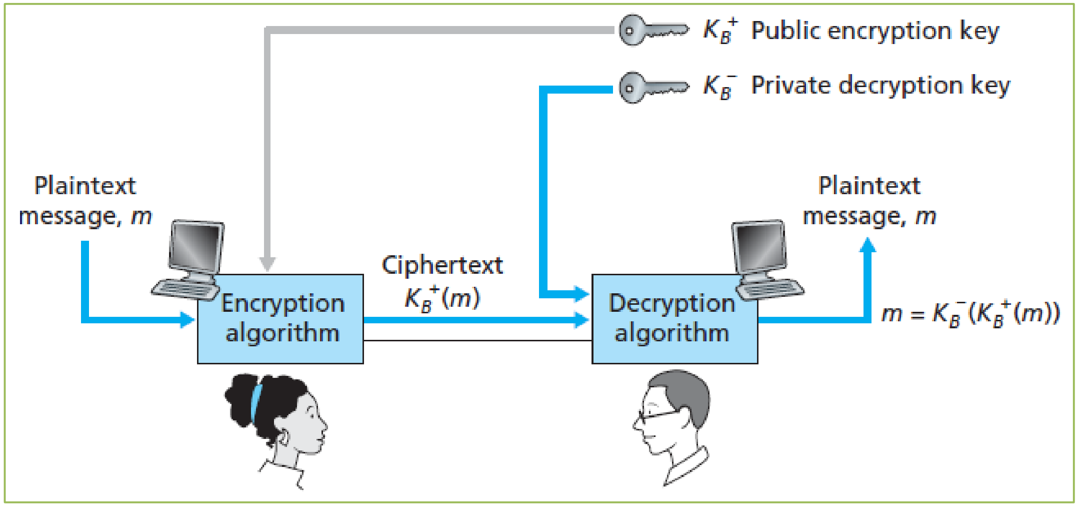
- Popular Symmetric Key Cryptography Algorithms:
  - ✓ Data Encryption Standard (DES)
  - ✓ Triple DES
  - ✓ Advanced Encryption Standard (AES)

- Popular Asymmetric Key Cryptography Algorithms:
  - ✓ RSA Algorithm
  - ✓ Diffie-Hellman Key Exchange (DHK)
  - ✓ Elliptic Curve Cryptography (ECC)

Confidentiality

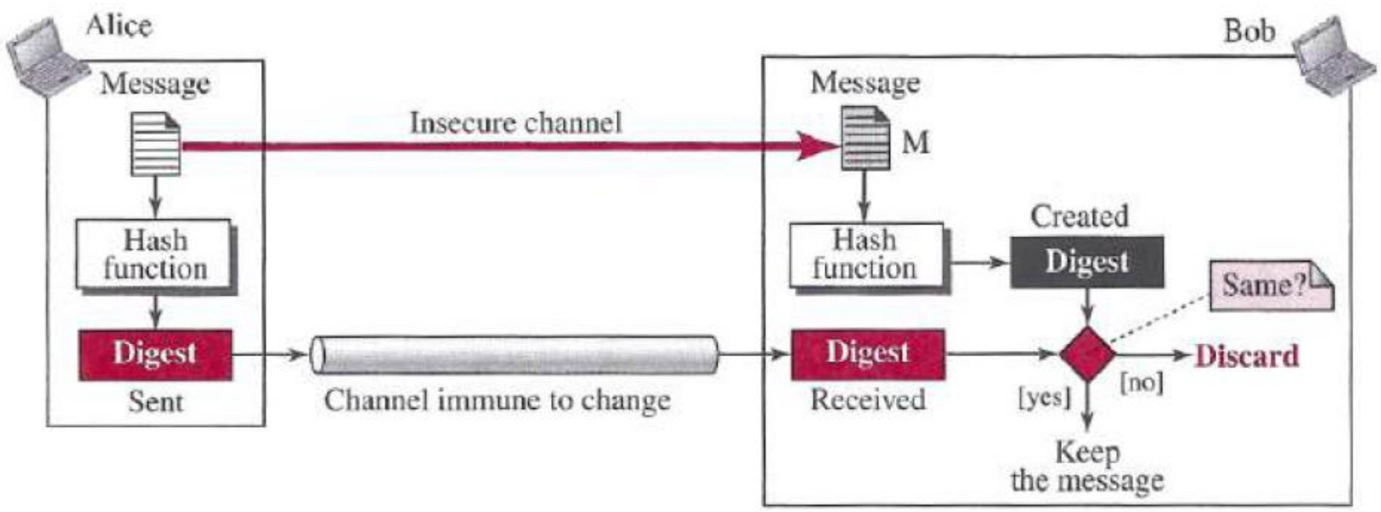
- Only the sender and intended receiver should be able to understand the contents of transmitted message.

- Primary objective of Symmetric and Asymmetric Key cryptographic algorithms is to provide confidentiality.



Message Integrity

- We must ensure message integrity: the message should remain unchanged.
- One way to preserve the integrity of a document is through the use of a *fingerprint*.
- The electronic equivalent of the document and fingerprint pair is the *message* and *digest* pair.



- In general, message Digest is generated by a hash function

Hash Function

- Cryptographic Hash Function is required to have the following properties:
  - takes an input, *m*, and computes a fixed size string *H(m)* known as a hash
  - It is computationally infeasible to find any two different messages *x* and *y* such that *H(x) = H(y)*

- Popularly used Hash Functions:
  - MD5
  - SHA-1

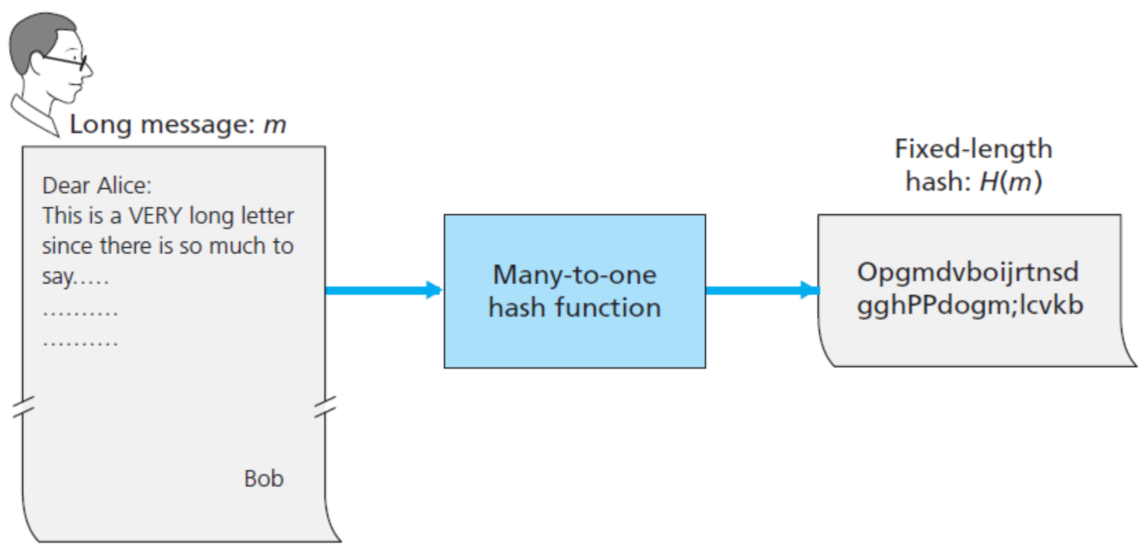
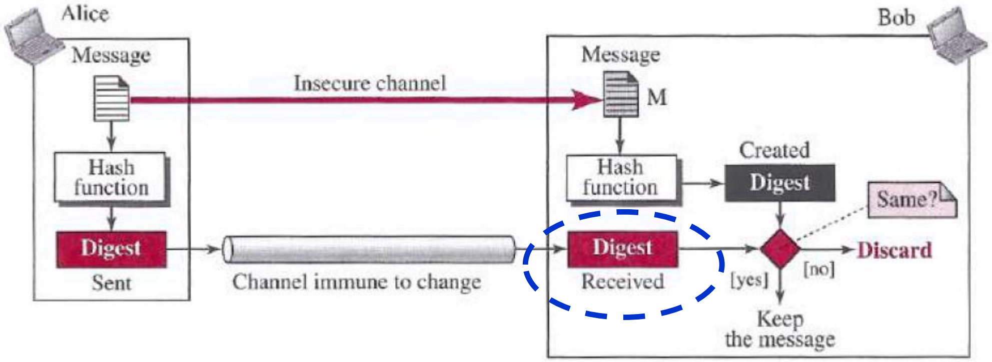


Figure 8.7 ♦ Hash functions

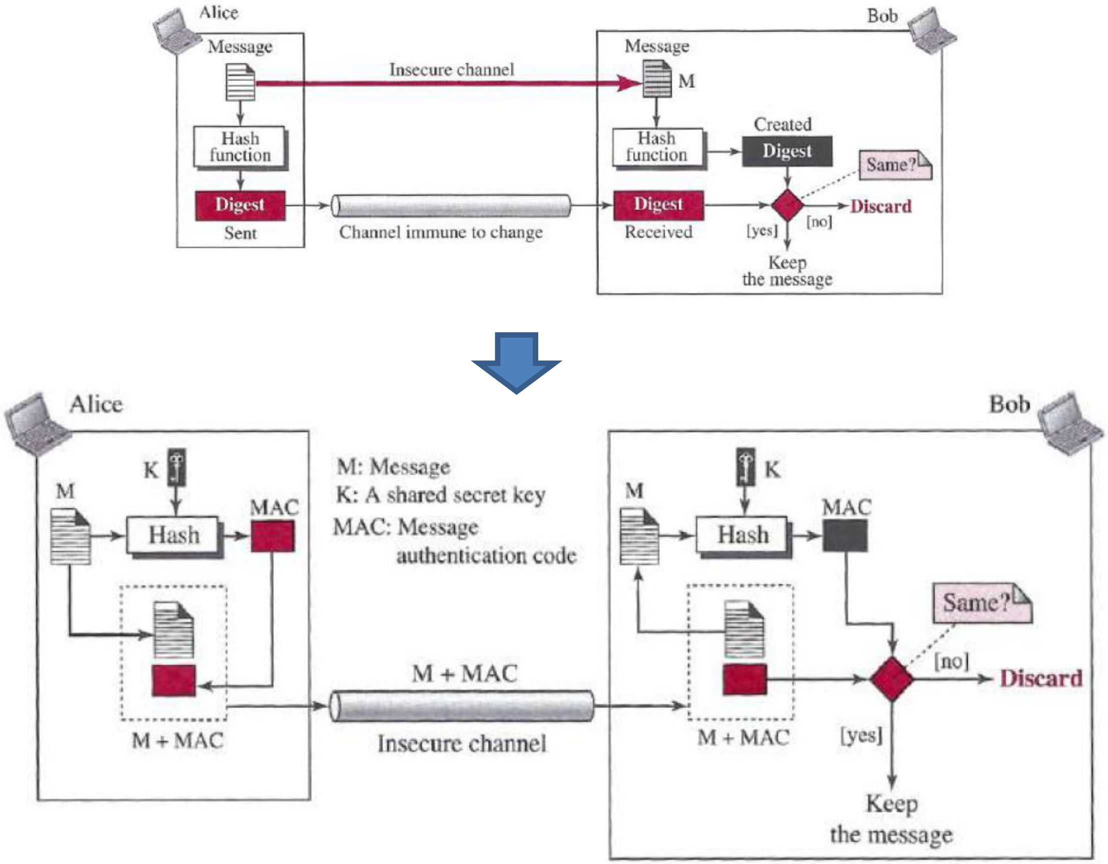
Message Authentication



- A digest can be used to check the integrity of a message.
- But, how to ensure the authentication of the message - the message indeed originated from Alice

- **Solution**: we need to include a secret shared between Alice and Bob.
  - This shared secret, which is nothing more than a string of bits, is called the **authentication key**.
  - Message digest of message and authentication key is called Message Authentication Code (**MAC**).

Cont…



- One nice feature of a MAC is that it does not require an encryption algorithm
- Note: MAC => Message Authentication Code / Message Integrity Code (MIC).
  MAC Layer => Medium Access Control Layer

Digital Signature

- Digital signature is a type of electronic signature based on Asymmetric Key cryptography.
- Purpose is to validate the authenticity and integrity of digital documents, messages, or software.

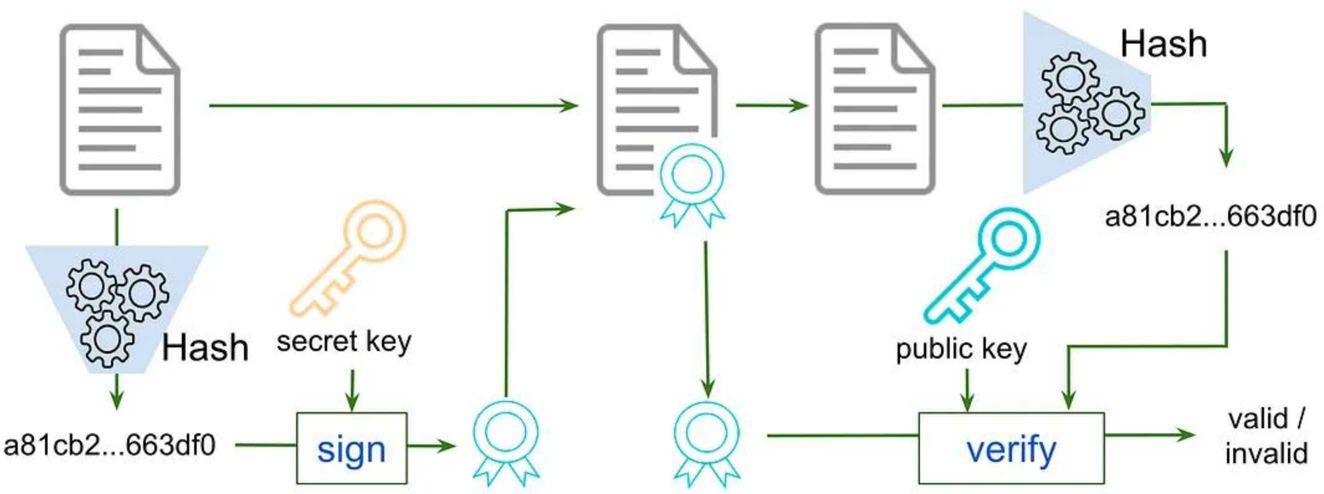


Image Source: https://kyle-crypto.medium.com/digital-signature-c601da3c0c02

Public Key Infrastructure (PKI)

- For public key cryptography to be useful, we need to be able to verify that we have the actual public key of the entity with whom we want to communicate.
- PKI makes the public key trustable. PKI uses trusted Certification Authority and Public Key Certificates.

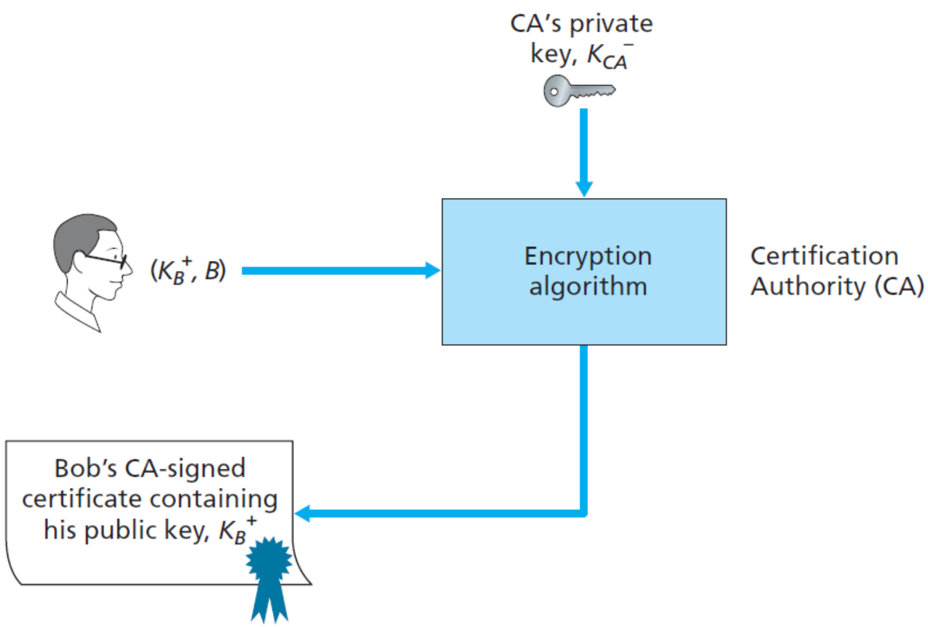


Image Source: J.F. Kurose and K.W. Ross, "Computer Networking – A Top-Down Approach", 8th Ed, Pearson India Education, 2023.

End-Point Authentication

- It is the process of one entity providing its identity to another entity over a computer network.
- Authenticating a “live” party, at the point in time when communication is actually occurring.
- Solution #1: Simplest one is party #1 sends its ID/name to party #2. ⇐ Not Secured
- Solution #2: Mostly used one is party #1 uses its user ID and password in authentication. ⇐ Not Secured, as password can be sniffed
- Solution #3: Lets assume both the parties have a secret key K. Then, party #1 can encrypts its user ID and password and send to part #2 for authentication. ⇐ Not Secured, as replay attack is possible
- Solution #4: Use nonce along with password encryption.

IoT PHY Layer (1/2)

- IEEE **802.15.4 PHY** manages
  - Physical RF transceiver of the sensing device, Channel selection,
  - Energy management, and Signal management

- Standard supports multiple channels (e.g. 16 channels in the 2.4 GHz ISM radio band)

- Reliability is ensured by modulation techniques (at PHY layer):
  - Direct Sequence Spread Spectrum (DSSS), Direct Sequence Ultra-Wideband (UWB), and Chirp Spread Spectrum (CSS).
  - They achieve reliability by occupying
    - more bandwidth at a lower power spectral density (in W/Hz) in order to achieve less interference along the frequency bands,
    - together with an improved Signal to Noise (SNR) ratio at the receiver.

- PHY data frames occupy at most 128 bytes
  - packets are *small* in order to minimize the probability of errors take place in low-energy wireless communication environments
    - <SYNC Header 5 byte> <PHY Header 1 byte><PHY Payload 127 byte>

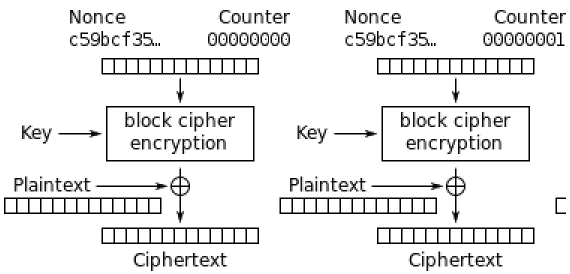
IoT MAC Layer (2/2)

- IEEE **802.15.4 MAC** manages
  - accesses to the physical channel
  - network beaconing
  - validation of frames
  - guaranteed time slots (GTS)
  - node association/joining
  - link level security
  - data services
- Types of devices based on the capabilities and roles in the network
  - FFD
  - RFD
- Types of frames
  - Data
  - ACK
  - Beacon
  - Command
    - Association request, Association response, Disassociation notification, Data request, Beacon request, GTS request, etc.
- IEEE 802.15.4 can support network topologies
  - peer-to-peer,
  - star,
  - cluster networks
- Devices are identified using
  - 16-bit short identifier
    - usually employed in restricted environments
  - 64-bit EUI-64
- Collisions during data communications are managed by CSMA/CA
- IEEE **802.15.4e** devices are
  - synchronize to a slot frame structure
  - Slotframe is a group of slots repeating over time
- Time synchronization is done using
  - acknowledgment-based, or
  - frame-based method

Security in IoT PHY & MAC

- *Security Suite in MAC:*
  - IEEE 802.15.4 standard support various *security suite* at the MAC layer,
  - *Security suites* are distinguished by
    - security guarantees provided, and size of message integrity data employed

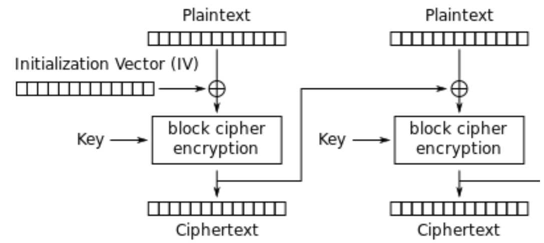
CTR Mode



Security suite includes security mechanisms defined for MAC layer frames

Name	Description
Null	No security
AES-CTR	Encryption only, CTR Mode
AES-CBC-MAC-128	128 bit **MAC**
AES-CBC-MAC-64	64 bit **MAC**
AES-CBC-MAC-32	32 bit **MAC**
AES-CCM-128	Encryption & 128 bit **MAC**
AES-CCM-64	Encryption & 64 bit **MAC**
AES-CCM-32	Encryption & 32 bit **MAC**

CBC Mode



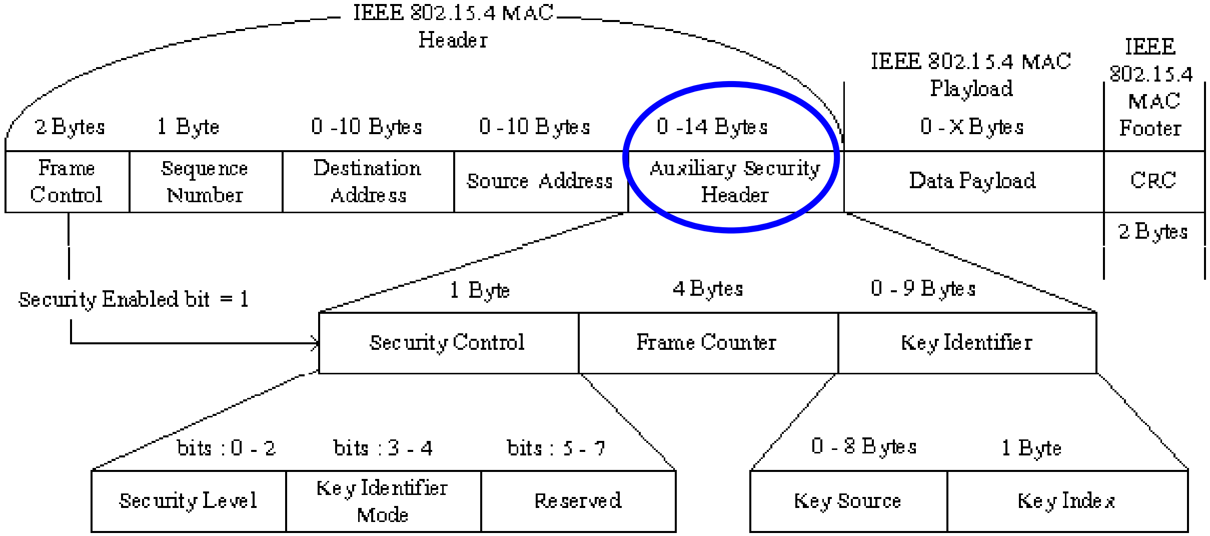
AES: Advanced Encryption Standard MAC: Message Authentication Code
CBC: Cypher Block Chaining CTR: Counter Mode CCM: CTR + CBC

Achieved Security Requirements

Security Suites	Achieved Security Requirements	Remarks
AES-CTR	Confidentiality	
AES-CBC-MAC-32, AES-CBC-MAC-64, AES-CBC-MAC-128	*Data Authenticity, Integrity,*	MAC/MIC is created from 802.15.4 MAC layer header plus the payload data
AES-CCM-32, AES-CCM-64, AES-CCM-128	*Confidentiality, Data Authenticity, Integrity*	

AES: Advanced Encryption Standard MAC: Message Authentication Code
CBC: Cypher Block Chaining CTR: Counter Mode CCM: CTR + CBC

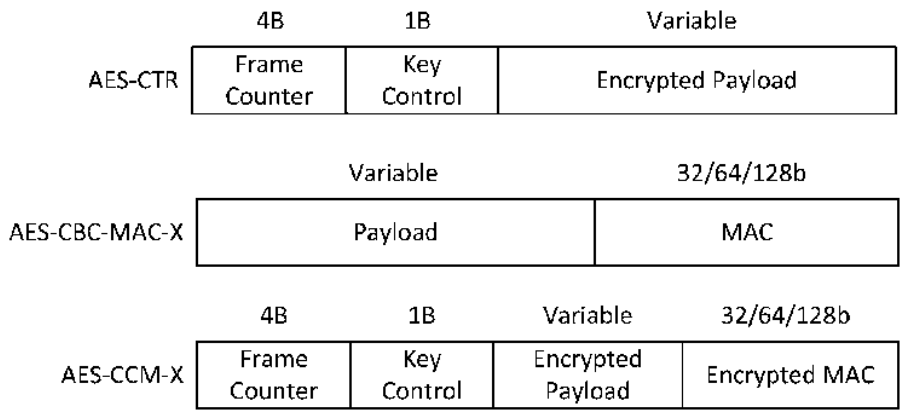
Data Frame with Security Header



- *Security Enabled Bit* field of the *Frame Control* field being set
- *Auxiliary Security Header* is employed only when security is used
  - ***Security Control*** field identifies the *Security Level* mode
  - ***Frame Counter*** used to ensure sequential freshness of frames sent by this device.
  - Standard employs 128-bit keys that is known implicitly by the two communication parties, **OR** determined from information transported in the ***Key Identifier*** field.
    - *Key Source* subfield specifies the group key originator
    - *Key Index* subfield identifies a key from a specific source

Security-related information

- The various *security suites* require transportation of <u>security-related information</u> for different configurations



Security Suites	Achieved Security Requirements
AES-CTR	Confidentiality
AES-CBC-MAC-X	*Data Authenticity* *Integrity*
AES-CCM-X	*Confidentiality* *Data Authenticity* *Integrity*

- <u>Protection against</u> message replay attack:
  - ✓ *Frame Counter* and *Key Control* fields are commonly used to achieve the protection in all security modes
  - ✓ *Frame Counter* sets the unique message ID by the sender
  - ✓ The *key counter* (Key Control field) is incremented if the maximum value for the *Frame Counter* is reached.

Cont…

- To support semantic security and also replay protection,
  - ✓ each block is encrypted using a different *nonce* or *Initialization Vector* (IV).

- Initialization Vector (IV) can be
  - ➢ a pseudorandom number,

  OR
  - ➢ [Frame Counter (4B), Key Counter (1B), Flags (1B), sender's address (8B), Block Counter (2B)]

    - ❖ *Block counter*: The sender breaks the original packet into 16-byte blocks, with each block identified by its own block counter.

Access Control

Access Control Mechanisms:

- ✓IEEE 802.15.4 MAC also provides access control functionalities

- ✓The 802.15.4 radio chips of the device stores an access control lists (ACL)
  - maximum of 255 entries (generally for 255 destination address)

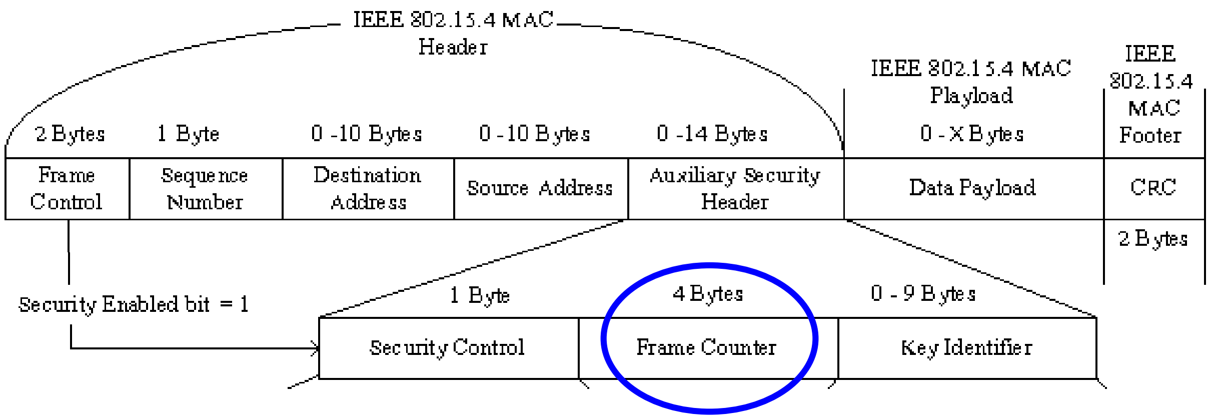
- ✓Each ACL entry stores
  - an IEEE 802.15.4 ***address***,
  - a ***Security Suite*** *identifier* field
  - the security material required to process security
    - cryptographic ***Key*** for suites supporting encryption,
    - the *Nonce* or **IV**,
    - the most recently received packet's identifier in the ***Replay Counter*** field

Address	Security Suite	Key	Last IV	Replay Counter

Added Security with TSCH

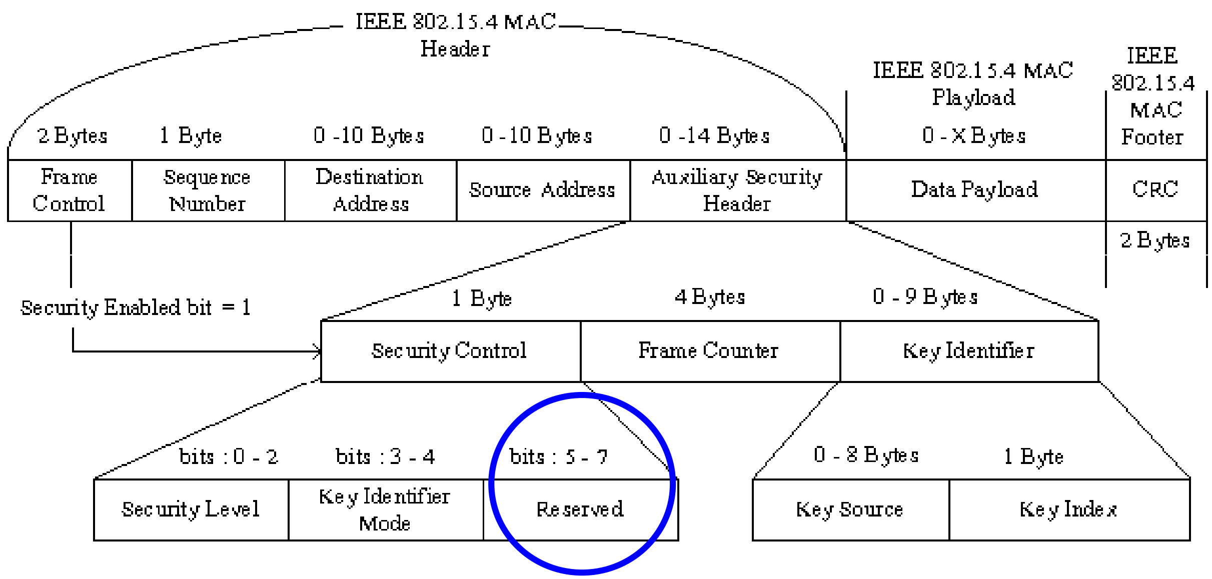
- IEEE 802.15.4 - 2011 provides security services at the MAC layer
- IEEE 802.14.4e - 2012 introduces few modifications in security services corresponding to different modes

- Addendum defines the possibility of using **null** and **4 or 5-byte** *Frame Counter* values
  - Frame counter is used to ensure sequential freshness of frames sent by this device.
  - It can be set to the global Absolute Slot Number (ASN) of the network
    - usage of the ASN as a global frame counter value enables
      - time-dependent security,
      - replay protection and
      - semantic security.



Cont…

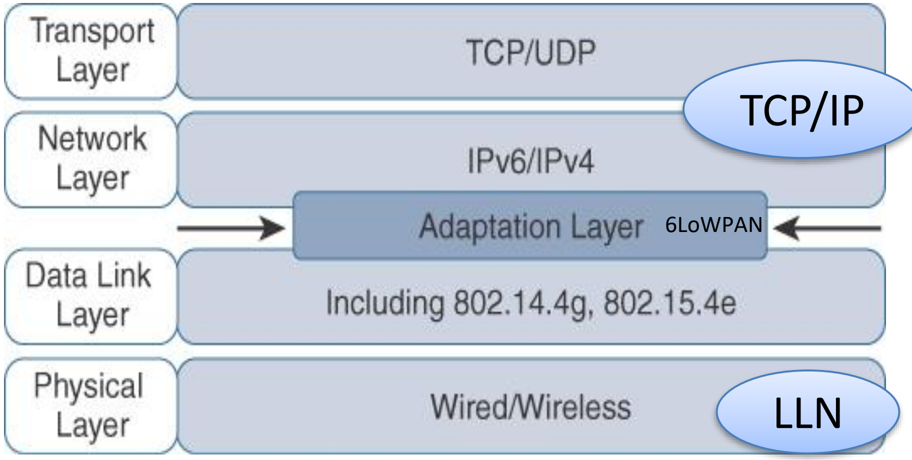
- IEEE 802.15.4e MAC frame employs two bits from the reserved space of *security control* filed:
  - bit 5 to enable suppression of the *Frame Counter* field i.e. null
  - bit 6 to distinguish between a *Frame Counter* field occupying 4 or 5 bytes



Adaptation Layer Protocol

- 6LoWPAN is currently a key technology to support Internet communications in the IoT

- It defines:
  - Header compression
    - Compress the NET and TRN layer headers
  - Neighbor discovery
  - Address auto-configuration



- It supports four header types:
  - Not 6LoWPAN
  - Dispatch
  - Mesh Addressing
  - Fragmentation

Security in 6LoWPAN

- No security mechanisms are currently defined in the context of the 6LoWPAN adaptation layer

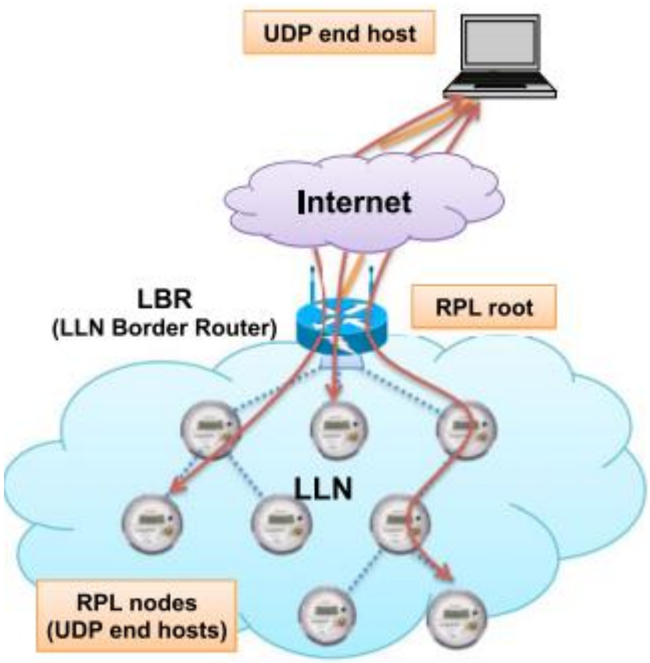
- But many relevant documents include discussions on the security vulnerabilities, requirements and approaches to consider for the usage of network layer security

  - ✓ RFC 6606: identifies the importance of addressing security and the usefulness of AES-CCM available at the hardware of IEEE 802.15.4 sensing platforms.

  - ✓ RFC 3971: SEcure Neighbor Discovery (SEND)

  - ✓ RFC 3972: Cryptographically Generated Addresses

RPL



RPL protocol supports many **control messages**

- DIO (DODAG Information Object)
- DIS (DODAG Information Solicitation)
- DAO (Destination Advertisement Object)
- DAO-ACK (DAO acknowledgment)
- CC (Consistency Check)
- Etc.

- RPL builds a Destination Oriented Directed Acyclic Graph (DODAG) **topology**
  - ✓ Each DODAG starting from root has unique ID
  - ✓ Parameters used: link costs, node attributes (e.g. rank), node status information
  - ✓ Function: objective function (OF).

Security in RPL

- RPL specification defines
  - secure versions of the various routing control message exchange
  - three **security modes**
    - Unsecured
    - Preinstalled
      - using a pre-configured symmetric key
      - Achieve: Confidentiality, Integrity, Data authentication
    - Authenticated
      - A device may initially join the network using a pre-configured key, and then obtain a different key from the key authority with which it may start functioning as a router.

RPL Control Message

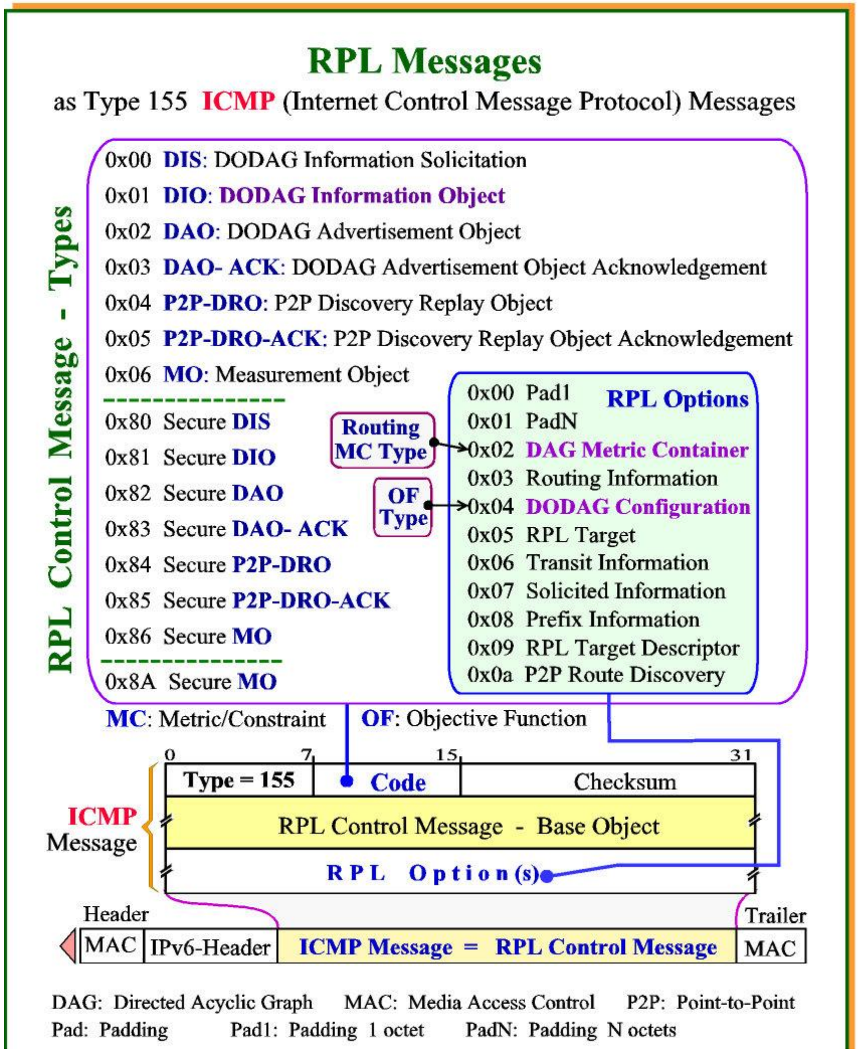


Image source: https://www.researchgate.net/publication/326960497_RPL_messages_and_their_structure

Security in RPL

1B	1B	2B
Type	Code	Checksum
Security		
Base		
Option(s)		

Fig: **Security fields** after the 4-byte ICMP header for a secure RPL control message

1b	7b	1B	2b	3b	3b	1B
T	Reserved	Algorithm	KIM	Resvd	LVL	Flags
Counter						
Key Identifier						

Fig: Format of the **Security** field

T: indicate the type of counter filed – timestamp or incremental counter value

Algorithm: specifies the encryption, MAC, and signature scheme the network uses

KIM: Key Identifier Mode

-- indicates whether the key used for packet protection is determined implicitly or explicitly and

-- indicates the particular representation of the Key Identifier field

LVL: Security Level

-- indicates the provided packet security and allows for varying levels of data authentication and, optionally, of data confidentiality

Flags: Reserved Flag Fields

Counter: can represent counter or timestamp

-- Semantic security and protection against packet replay attacks is provided with the help of the Counter field

Key Identifier:

-- indicates which key was used to protect the packet

RFC 6550 **(RPL)** defines all the fields in details.

Security in RPL

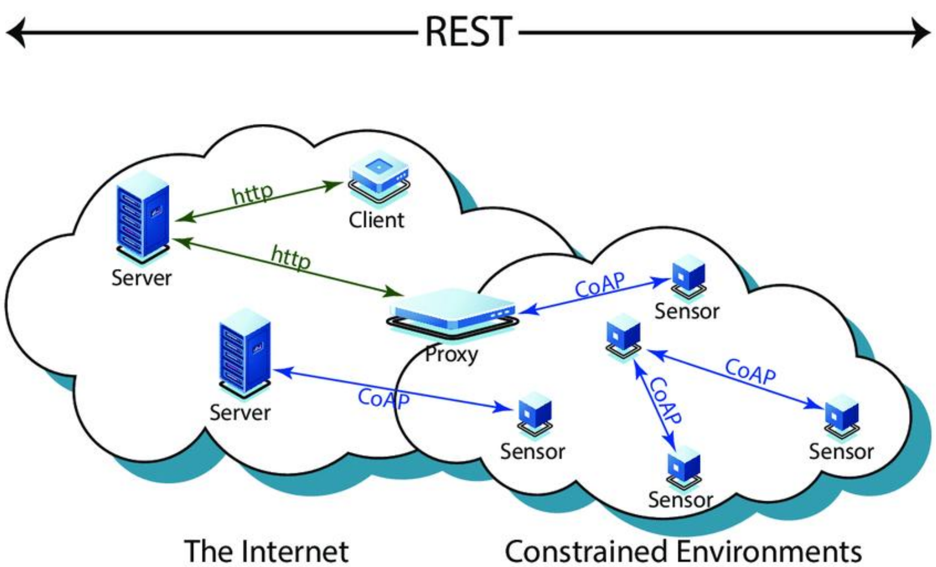
- Support of Confidentiality & delay protection
  - Using AES-CCM(Counter with CBC-MAC)
  - MAC is applied on entire IPV6 packet

- Support of Integrity and Data Authenticity:
  - Using AES-CCM-128
  - OR, using RSA with SHA-256

- Support of Semantic Security and Protection Against Replay Attacks:
  - It is provided with the help of the *Counter* field

- Support for Key Management:
  - The KIM (Key Identifier Mode) field of the *Security* section illustrates different key management approaches

CoAP



CoAP: Constrained Application Protocol

- For constrained and low-power networks.
- Follows the request-response messaging pattern
- The request is sent using Confirmable (CON) or Non-Confirmable (NON) message.
  - Uses stop-and-wait mechanism with exponential backoff to achieve reliability
- Response is sent using several scenarios (e.g. piggy-backed, separate response, etc)

Other Features:

- Very efficient RESTful protocol (i.e. uses REST: Representational State Transfer architecture)
- Low header overhead and parsing complexity
- Uses both asynchronous & synchronous messaging
- Mainly uses for UDP communications

CoAP Security

- CoAP Protocol defines bindings to DTLS (Datagram Transport-Layer Security) to transparently apply security to all CoAP messages

- Security Modes in CoAP:
  - *NoSec*
    - CoAP messages are transmitted without security applied.
  - *PreSharedKey*
    - sensing devices that are pre-programmed with the symmetric cryptographic keys required to support secure communications
  - *RawPublicKey*
    - A given device must be pre-programmed with an asymmetric key pair
    - supports authentication based on public-keys
    - The device has an identity calculated from its public key, and a list of identities and public keys of the nodes it can communicate with.
    - This is mandatory for implementing CoAP
  - *Certificates*
    - The device has an asymmetric key pair with an X.509 certificate
    - supports authentication based on public-keys

CoAP Security

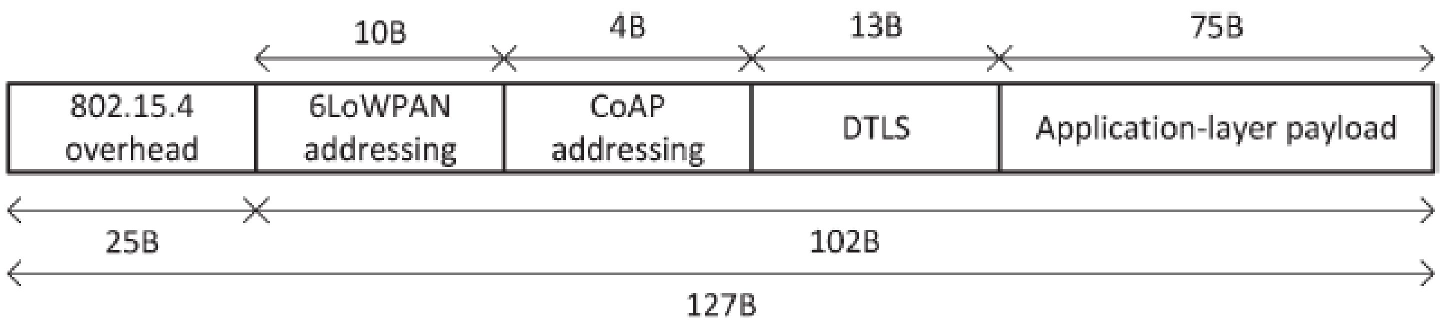


Fig: Payload space with DTLS on 6LoWPAN environments

- Elliptic Curve Cryptography (ECC) is adopted to support the ***RawPublicKey*** and ***Certificates*** security modes
  - ✓ ECC supports device authentication using the Elliptic Curve Digital Signature Algorithm (ECDSA), and also key agreement using the Elliptic Curve Diffie-Hellman Algorithm with Ephemeral keys (ECDHE).
- ***PreSharedKey*** security mode requires the TLS_PSK_WITH_AES_128_CCM_8 suite, which supports authentication using pre-shared symmetric keys and 8-byte nonce values, and encrypts and produces 8-byte integrity codes.

View from Operational Level

- Security Requirements from three Operation Levels:
  - Information Level
    - Integrity: received data should not been altered during the transmission
    - Anonymity: identity of the data source should remain hidden
    - Confidentiality: Data cannot be read by third parties
    - Privacy: The client's private information should not be disclosed
  - Access Level
    - Access Control: only legitimate users can access to the devices and the network for administrative tasks
    - Authentication: checks whether a device has the right to access a network and vice-versa
    - Authorization: ensures that only the authorized devices /users get access to the network services or resources.
  - Functional Level
    - Resilience: refers to network capacity in case of attacks and failures
    - Self Organization: denotes the capability of an IoT system to adjust itself in order to remain operational even in case of partial failure or attack

Security Mechanisms for IoT

- **Standard security mechanisms** that have been designed to satisfy the security requirements for IoT Services.
  - Encryption:
    - Standard Mechanisms:
      - Symmetric mechanism: AES-128, AES-192, AES-256 with CTR or CBC or CCM mode
      - Asymmetric mechanism: RSA , Elgamal
    - Above mechanisms mainly used for confidentiality
    - Authentication and Integrity protection are provided by Message Authentication Code (Symmetric Mechanism), Digital Signature (Asymmetric Mechanism) and Hash Functions.
      - Tag is computed and concatenated with message
      - Tag is obtained by AES-CBE-MAC (Symmetric scheme), Digital Signature Algorithm (Asymmetric scheme) , Elliptic Curve DSA (ECDSA), RSA with Hash Function e.g. MD5, SHA-160, SHA-256, SHA-512
  - Lightweight Cryptography:
    - Block cipher : PRESENT, CLEIFA, PRINCE
    - Hash function: PHOTON, SPONGENT

Security Mechanisms for IoT

- **Standard security mechanisms** that have been designed to satisfy the security requirements for IoT Services.

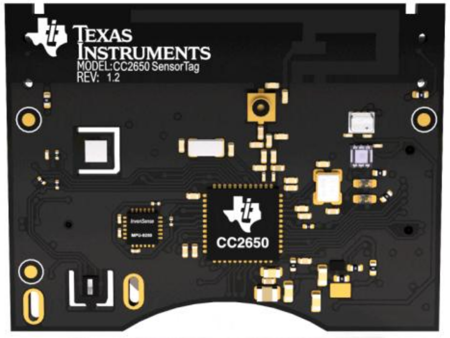
  - Secure Hardware
    - illegitimate user can perform side channel attacks as devices can be deployed in remote areas with less protection
    - It is possible to exploit both hardware and software solutions to eliminate or to mitigate
    - e.g. PUF – Physically Unclonable Functions
      - The basic concept of PUF is to exploit little differences introduced by the fabrication process of the chip to generate a unique signature of each device.
    - Shortcomings: increase of *power consumption* of the device

  - Intrusion Detection Systems
    - Complex anti-virus software and traffic analyzers cannot be used in IoT devices
    - Lightweight IDS: anomalies in system parameters, like CPU usage, memory consumption, and network throughput, may be indicative of an ongoing attack

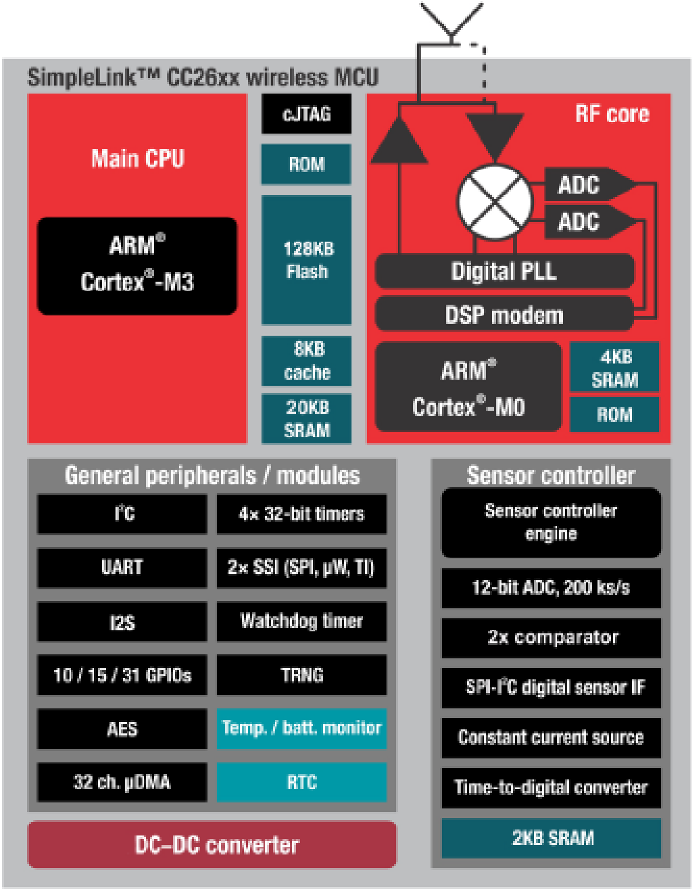
Implementation in Commercial Device

- In general, IoT devices include at least two microcontrollers:
  - ✓ one responsible for the management and processing of the data
  - ✓ other for connectivity

- Most of the IoT devices use ARM microcontrollers of Cortex-M series
  - Require minimal costs, power, and size: M0, M0+, and M23
  - Offers balance between performance and energy efficiency: M3 and M4
  - high performance embedded applications: M7



TI CC2650 Sensor Tag



TI CC26xx Functional Block Diagram

Implementation in Commercial Device

- In general, IoT devices include at least two microcontrollers:
  - ✓ one responsible for the management and processing of the data
  - ✓ other for connectivity

- Most of the IoT devices use ARM microcontrollers of Cortex-M series
  - Require minimal costs, power, and size: M0, M0+, and M23
  - Offers balance between performance and energy efficiency: M3 and M4
  - high performance embedded applications: M7

- In general, Cortex-M family do not integrate
  - ✓ any hardware pseudorandom number generator,
  - ✓ any module supporting cryptographic algorithms such as AES

- Support for cryptographic algorithms is implemented via software or by dedicated co-processors

Example:

- ✓ TI CC2540 MCU provides an AES security co-processor for encryption and decryption
- ✓ TI SoC CC2538 implements AES in hardware

Thanks!

Important reference:

- Granjal *et al.*, "Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues" *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, 2015.