# Security Aspects and Efforts Towards Secure Internet of Things

Wan Fariza binti Wan Abdul Rahman[1], Aisha Hassan Abdalla Hashim[2], Md. Rafiqul Islam[2]
[1]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Malaysia (UiTM), Malaysia*.
[2]*Department of Electrical and Computer Engineering, Kuliyyah of Engineering,*
*International Islamic University of Malaysia (IIUM), Malaysia*.
*wfariza@kelantan.uitm.edu.my*

***Abstract*—Internet of Things (IoT) consists of wired and wireless devices, typically supplied with minimum physical resources including limited computational and communication resources. Most of the devices are distinguished by their low bandwidth, short range, scarce memory capacity, limited processing capability and other attributes of inexpensive hardware. The resulting networks are more prone to traffic loss and other vulnerabilities. One of the potential networking challenges is to ensure the network communication among these deployed devices remains secure at less processing and communication overhead, and small packet size. The purpose of this paper is to highlight possible security attacks in Low Power and Lossy Networks (LLNs) as identifying pertinent security issues is an initial step to design the effective countermeasures. The IETF efforts in relevance to security implementation of this type of network are presented with focus on layer-2 and authentication mechanism at upper layer.**

***Index Terms*—IoT; 6LoWPAN; LLN; Security; Secure Routing; Authentication; DoS; Attacks.**

## I. INTRODUCTION

Internet of Things (IoT) is based on the smart objects working together through Internet Protocol (IP) connectivity. These objects are devices typically embedded with sensors, connected to the Internet, thus allowing them to be assessed, controlled and managed, regardless of their location, from anywhere, at anytime. The IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) refers to a network formed by this kind of devices that are compatible with the IEEE 802.15.4 standard.

IoT may carry sensitive information and require a high level of security support where the availability, integrity and confidentiality of data are of prime relevance. The embedded devices in public areas with weak wireless links are exposable to malicious entities exploit, and a mass surveillance, tracking and profiling of the users' movements and activities [1]. The physical exposure of the nodes allows an adversary to capture, clone, or tamper with these devices [2].

Secure transmission of routing messages is required to avoid disruption from attackers, which may degrade the overall performance of the network significantly. In non-secure routing, the network may be contaminated with false routing information, resulting in routing inconsistencies. A malicious node can perform packet snooping and launch replay attacks on other nodes. Attacks can also be done by sending broadcast messages, redirecting routes and so on, in order to drain the batteries of other nodes. The detection and treatment of attacks however, could be delayed by the multi-hopping process [2].

Compared to traditional networks, IoT poses new security challenges due to the following reasons:

- The highly distributed nature of IoT needs to achieve full interoperability between interconnected devices through adaptation and autonomous behavior. This requires a high degree of smartness and at the same time guaranteeing trust, security, and privacy of the users and their exchanged data.
- The deployment of IoT devices in unattended or remote locations with limited physical security lowers the barrier of accessing the data or security material stored on the devices through physical means [3].
- In many cases, the IoT may be deployed as collections of identical or near identical devices. Thus, security protocols should be designed to avoid a compromise of a single device from causing a compromise of the entire collection [4].
- To conserve energy, IoT devices may sleep for a long period and are unable to communicate during this time. Static security configuration does not suffice due to dynamicity (including mobility) of the IoT nodes in term of topology and node memberships. Time synchronization, self-organization and secure localization for multi-hop routing are critical to support [5].
- Layered security solutions (in which each layer takes care of its own security needs) fit well with traditional computer networks. However, limited energy, memory and processing resources of IoT devices require the security protocols to be more interconnected across layers to ensure efficiency [6].
- IP-only security solutions may not suffice in many IoT scenarios. The wireless medium used is broadcast in nature and anybody on the right frequency is able to overhear and even inject packet at will. Thus, protection of lower protocol layers is needed to ensure resistance against routing attacks [6]. Authenticated broadcast (and multicast) and bidirectional link verification may be necessary for secure routing protocol operation [5].
- The disadvantage of security gateway and IPsec tunnel mode in term of larger header size is significant at the 6LoWPAN frame maximum transfer unit (MTU) [7].

The IEEE 802.15.4 MAC provides an Advance Encryption Standard (AES)-based security mechanism whereby in conjunction with IPsec [8] can be used to obtain the intended security. However, in the worst case in terms of overhead, the mechanism consumes 21 bytes of MAC payload (21 bytes of

overhead in AES-(Counter with Cipher Block Chaining (CBC)-Message Authentication Code)-CCM-128 case, versus 9 and 13 for AES-CCM-32 and AES-CCM-64, respectively). The IEEE 802.15.4 standard specifies an MTU of 127 bytes, yielding 81 bytes of actual MAC payload with security enabled [9]. In order to avoid packet fragmentation and reassembly overhead, the size of messages in 6LoWPAN should not exceed a single IEEE 802.15.4 frame [2].

Considering the power constraints and limited processing capabilities of IEEE 802.15.4-compliant devices, IPsec which provides privacy and authentication service at the IP layer is computationally expensive. IPsec requires the Authentication Header (AH) for authenticating the IP header and the Encapsulating Security Payload (ESP) for authenticating and encrypting the payload. Two main issues of using IPsec for 6LoWPAN are processing power and key management. 6LoWPAN devices do not process huge amounts of data as well as do not communicate with many different nodes (due to bandwidth scarcity). In addition, IPsec requires two communicating peers to share a secret key that is typically established dynamically with Internet Key Exchange (IKEv2) [2].

IKEv2 [10], a component of IPsec, uses public-key identities to authenticate the initiator of a connection. Since these identities could easily be traced, the IKEv2 transmits this information in an encrypted packet [6]. The exchange of IKEv2 packets also incurs additional packet overhead. Thus, IKEv2 will not work well in 6LoWPAN due to the purpose of minimizing the amount of signalling in 6LoWPAN [2].

Since a 6LoWPAN node is incapable of operating all IPsec algorithms on its own, a 6LoWPAN requires its own keying management method with minimum overhead in packet size and in the number of signalling messages exchanged. Link-layer encryption and authentication may not be sufficient to provide confidentiality, authentication, integrity and freshness to both data and routing protocol packets. Time-synchronization, self-organization, and secure localization for multi-hop routing need to be supported [2]. The design of IoT security protocols need to consider its nature that relies on lossy and low-bandwidth channels for communication between resource-constraint nodes in term of CPU, memory and energy [6].

The following sections highlight the important security elements that need to be maintained in 6LoWPAN and related threats/attacks. Besides, the efforts and approaches proposed by IETF in achieving security in IoT will be presented as well.

## II. IoT Security Properties

### A. Availability

Availability can be defined as maintaining efficient and correct operation of routing and neighbour discovery exchanges (including the requested information), as well as forwarding services when they are required for the functioning of the serving network. Due to its low-power, tight memory and limited computation nature, 6LoWPAN are vulnerable to Denial-of-Service (DoS) attacks. DoS attacks are launched to deplete a node's resources by continuously sending requests to be processed or introducing a high power jamming signal that makes the nodes dysfunctional. Network availability can also be disrupted by flooding the network with a large number of packets [6, 11].

Overload attacks are a form of DoS attacks whereby a malicious node overloads the network with irrelevant traffic. As a result, the nodes' energy (especially those rely on batteries or energy scavenging) draining more quickly. This significantly shortens the lifetime of networks of energy-constrained nodes [3].

Overload attacks can be countered by some security measures such as by introducing quotas on the traffic rate that can be sent by each node, allowing only trusted data to be received and forwarded, and isolating nodes that send traffic above a certain threshold based on system operation characteristics [3].

### B. Confidentiality

The term confidentiality involves the protection of routing information as well as routing neighbour maintenance exchanges so that only authorized and intended network entities can view or access it. Further, confidentiality also extends to the neighbour state and database information within the routing device [3].

Only authenticated users must be able to access and handle the data. Without proper security measures, confidential information might be snooped by "man-in-the-middle" (MITM). The attacker might modify or introduce data packets in the network [11].

### C. Authentication, Integrity & Access Control

Another requirement for secure communication is mutual authentication of the routing peers (node authentication) prior to exchanging route information as well as ensuring that the source of the route data is from peer. Node authentication should be supported by lightweight security solution to guarantee message integrity and prevent misbehaving nodes participation in the network. Thus, a node must authenticate itself to trusted nodes before participating in the network [3].

Upon establishing a secure communication channel, digital certificates and secret keys are used for authentication purpose. The certificates should be validated prior to use of the associated keys to counter potential resource overloading attacks. The receiving nodes that validate signatures and sending nodes that encrypt messages should also be cautious of cryptographic processing usage. This is due to the reason that these processes require resources and could also be exploited for attacks [3].

Integrity on the other hand, refers to the protection of routing information and derived information maintained in the database, from unauthorized modifications, insertions, deletions, or replays. Integrity is related with the access control which provides protection against unauthorized use of the network asset and deals [3].

## III. Security Threats and Routing Attacks

A threat is a potential violation of security which exists when there is a circumstance, capability, action, or event that could breach security and results in harm. Compared to threat, security attack is a deliberate attempt using certain technique to evade security services and violate the security policy of a system [3].

### A. Attacks in General

A device in a network may be susceptible to MITM or eavesdropping attacks, especially if operational keying materials, security parameters, or configuration settings, are exchanged in clear via wireless links. Once the keying material has been obtained, the attacker might be able to

recover the secret keys established between the communicating entities. Thus, the authenticity and confidentiality of the communication channel, as well as the authenticity of commands and other traffic exchanged over this communication channel, will be compromised. Eavesdropping may occur if the communication channel is not sufficiently protected, or in the event of session key compromised due to a long period of usage without key renewal [6]. One way to counter this kind of attack is through the use of data encryption for all routing exchanges. The implementation of CCM mode AES-128 bit method is believed to be secure against a brute-force attack [3].

Further, devices typically do not have prior knowledge about each other, and incapable of differentiating friends and foes via completely automated systems. The key establishment protocol (which provides cryptographic device authentication) need to be complemented with a human-assisted authorization step. Thus, MITM attack can also happen in this way [6].

Public key cryptography (PKC) primitives are typically avoided due to relatively heavyweight conventional encryption methods [2]. Small CPUs and scarce memory limit the usage of resource-expensive cryptoprimitives (such as PKC) as used in most Internet security standards [6].

### *B. Routing Attacks*

Routing information can be spoofed, altered, or replayed in order to create routing loops, attract/repel network traffic, extend/shorten source routes and so on. A few types of common routing attacks are Sinkhole attack, Selective Forwarding, Wormhole attack, and Sybil attack [6].

Sinkhole attack (also known as blackhole attack) is a situation where an attacker declares himself to have a high-quality route/path to the base station to convince other nodes to route packets through it. Consequently, the attacker can make use of all packets passing through it. In Selective Forwarding, the attacker may selectively forward packets or simply drop a packet [6]. If all packets are dropped, the attacker is also often referred to as a "blackhole". Selective Forwarding can be countered by some measures such as the usage of multipath routing of the same message over disjoint paths, or dynamically selecting the next hop from a set of candidates. Using multipath routing can guarantee that if a message gets lost on certain path due to Selective Forwarding attack, there will be another route that can still deliver the message. However, such approach is inherently suboptimal from an energy consumption point of view. On the other hand, dynamic selection of next hop routers involves a constantly changing topology [3].

In Wormhole attack, on the other hand, the attacker may record packets at one location in the network and tunnel them to another location in order to influence perceived network behavior. This can greatly affect the functionality of routing. However, the pure Wormhole attack is nearly impossible to detect. In Sybil attack, an attacker presents multiple identities to other devices in the network [6].

Another form of attacks in a Low Power and Lossy Networks (LLNs) is HELLO flood attacks which lead nodes to believe that suitable routes are available even though they are not and hence constitute a serious availability attack [3].

Nodes are required to broadcast HELLO packets to announce themselves to their neighbours. An attacker can launch a HELLO attack against RPL [12] by sending out (or replaying) DODAG Information Object (DIO) messages. Lower-power nodes might then attempt to join the Destination Oriented Directed Acyclic Graphs (DODAGs) at a lower rank than they would otherwise [3]. A node receiving such a HELLO packet may assume that it is within radio range of the sender. However, this assumption may be false if an attacker broadcasted routing or other information using large enough transmission power, which can convince every node in the network that the adversary is its neighbour. As a result, every node in the network could cause a large number of nodes to attempt to use the route advertised, thereby sending the legitimate packets beyond the actual destination [13].

The request (REQ) messages from an adversary transmitted with large power can be ignored if each sensor node constructs a set of reachable sensor nodes, and is only willing to receive the REQ messages from this set of neighbour nodes. Using this approach, the damage from a HELLO flood attack can be restricted within a small range. To defend against this attack, each REQ message forwarded by a node is encrypted by a key. The new encryption key is generated on-the-fly (i.e. during communication) for any two sensor nodes sharing some common secrets. Thus, any nodes reachable neighbours can decrypt and verify the REQ message while the attacker will not know the key [13].

### *C. Privacy Threat*

Privacy refers to the rules under which data referring to individual users may be accessed [14]. The issues of privacy may arise both during data collection and transmission. IoT devices such as sensors and actuators may involve data or processes belonging to individuals. Privacy protection is essentially important since the sensor data may be recorded continuously thus allowing significant information about an individual to be gathered from the sensor readings. Privacy protection should be considered for end-to-end confidentiality; otherwise intermediary nodes will learn the content of potentially sensitive messages sent between a client and a resource server [15].

## IV. SECURITY APPROACHES

### *A. 1Layer-2 Security*

IEEE 802.15.4 standard is one of the successeful enabling technologies for short-range low-rate wireless communications in which MAC packets are protected by means of symmetric-key cryptography techniques. Layer-2 security for IEEE 802.15.4e-based LLNs has been defined in [16] with some amendments to IEEE 802.15.4 standard. Time-slotted Channel Hopping (TSCH); a novel MAC protocol provides better supports for multi-hop communication through few upgrades of security-related aspects. 6TiSCH working group was created to define open standards in support of the adoption of IPv6 over TSCH mode of the IEEE 802.15.4e standard.

Among the security issues tackled by 6TiSCH working group are join processes, the keying material and authentication mechanism needed by a new node to join an existing network, and secure transfer of data between neighbouring nodes. Three classifications of possible secure network configurations are Fully Secure, Unsecure, Partial Secure, and Hybrid Secure networks [16].

A Fully Secure network is a network whereby all the devices in the network have already obtained all the required keys, and all packets are encrypted and authenticated using

specific keys, depending on the messages carried. In Unsecure network, the data encryption, the message integrity, and the peer authentication are not implemented. All the MAC frames are exchanged in clear. Even if any pair of nodes are capable of possessing security, they are not allowed to do so. In Partial Secure network, only the integrity of message is supported. On the other hand, in Hybrid Secure Network, there are still a group of nodes that have not yet authenticated by the network (due to incomplete join procedure) [16].

A layer-2 secure link among the nodes can be established through a set of consecutive sets (i.e. Setting-up, Bootstrap, Join and Key Negotiation phases). Two different types of layer-2 key are production network key and per peer L2 key. The production network key is a secret shared by all the authorized nodes which can only be obtained upon correctly completing a join process with authorization and authentication procedures. Instead, the per peer L2 key is negotiated only between a couple of nodes through a Key Management Protocol (KMP) strategy [16].

Another special layer-2 key, known as master L2 key, represents an initial secret. This key is shared among all the nodes and configured by the network administrator before the network deployment. However, protection of this key should be ensured through specific software-based and/or hardware-based mechanisms since an attacker may physically access and extract it [16].

The purpose of each L2 key is to protect a specific set of messages. The master L2 key is particularly used for protecting enhanced beacon (EB) and data frames exchanged during the join procedure. The production network key is used to protect broadcast messages and MAC frames exchanged during the Key Negotiation Phase. The per peer L2 key is used to encrypt and authenticate messages exchanged between two nodes at the MAC layer. The calculation of the per peer L2 key imposes to use the CCM algorithm and a 128-bit key to protect MAC frames [16].

As mentioned previously, a layer-2 secure link can be established through four phases; Setting-up, Bootstrap, Join and Key Negotiation phase. In Setting-up phase, all the required secrets to initialize a secure domain are stored into the devices. The secrets and parameters stored include the master L2 key, private key of the node, public key of the node stored within a certificate and the certificate of the certification authority.

The Bootstrap phase is used for initializing security MAC attributes with different implementation for both personal area network (PAN) coordinator and the join node. Secure bootstrapping refers to the process whereby a device securely joins the network at a given location and point in time. This covers the device authentication and authorization, as well as the transfer of security parameters for trusted network operation [6].

The Join Phase is handled by the upper layers, providing authorization and authentication services. Key Negotiation phase handles KMP to negotiate a layer-2 key between a pair of nodes that are directly connected at MAC layer [16].

In TSCH MAC, time synchronization and channel hopping information are advertised in EB frames to be used by nodes to determine the timeslots available for transmission and reception of MAC frames. An attacker can inject forged EB frames and can cause replay and DoS attacks to TSCH MAC operation. Thus, all EB frames must be integrity protected [17].

The join process involves a Joining Node and a Join Assistant (JA). The JA is part of the production network, and participates in one or more DODAGs, such that it is reachable from the 6LoWPAN Border Router (6LBR), and the Join Coordination Entity (JCE). Production network is a term used to refer to an 802.15.4e network whose encryption/authentication keys (network-wide group keys or per-link keys) are determined by some algorithm [18].

First, the Joining Node needs to find the Join Network by listening to an EB which are broadcast in designated slotframes by JA. The EB provides a way for the Joining Node to synchronize itself to the overall timeslot schedule. The Joining Node can send a Router Solicitation based on the Aloha period and can receive a Router Advertisement giving the Joining Node a prefix and default route to send join request [18].

The JCE must authenticate itself to the Joining Node so that the Joining Node will know that it has joined the correct network. Similarly, the Joining Node must authenticate itself to the JCE so that the JCE will know that this node belongs in the network. This two-way authentication occurs in the Datagram version of the Transport Layer Security/Constrained Application Protocol (DTLS/CoAP) session that is established between the JCE and the Joining Node [18].

The Joining Node sends traffic to the JA, which forwards it using the normal RPL DODAG upwards routes. By this way, the Joining Node will reach the JCE using regular routing. The DODAG does not have information about this node. To get connectivity from the JCE to the Joining Node, the JCE uses loose-source routes to address packets first to the JA, which will then forward to the Joining Node [18].

The join process must deal with three kinds of threats; (1) threats to the Joining Node, (2) threats to the resources of the network, and (3) threats to other Joining Nodes [18]. Threats to the Joining Node may occur when an attacker convinces the Joining Node that it is the legitimate network and the Joining Node will not know about this until the ClientCertificate from the JCE is obtained [18].

On the other hand, the Joining Node (nodes of malicious network) may also mount attacks on legitimate nodes which have not yet joined the network. The malicious node can do this by sending very long certificate chains to validate, or can just feed the Joining Node legitimate chains that it observed (and replayed) from the legitimate JCE. Unfortunately, when the Joining Node finds that the DTLS connection is invalid, it may significantly run batteries down [18].

Two important network resources that may be attacked by malicious Joining Node are energy/bandwidth, and memory for routing entries. A malicious Joining Node could send many Neighbour Solicitation (NS) messages (from many made up addresses) to the JA. As a result, the JA would send many NS messages to the 6LBR which consumes bandwidth and energy from the members of the network along the path to the 6LBR. This type of attack can be mitigated by putting limits to the total bandwidth available for joining process [18].

### *B. Upper Layer Security*

CoAP [19] protocol which is a Hyper Text Transfer Protocol (HTTP)-like resource access protocol and runs over User Datagram Protocol (UDP) has been designed for LLNs. The DTLS [20] is used to secure CoAP in the same way as Transport Layer Security (TLS) secures HTTP. The DTLS

protocol provides communications privacy for datagram protocols, whereby eavesdropping, tampering, or message forgery is prevented, allowing client/server applications to communicate securely.

Although the DTLS protects the entire CoAP message including header, options and payloads, it only protects data hop-by-hop, in which all intermediary nodes can modify the transmitted information. Such condition will put risks from a privacy and security perspective as the intermediaries are free to delete resources on sensors and falsify commands to actuators. Further, due to the handshake procedure, DTLS incurs a large overhead cost. Thus, considering secure objects instead of secure session can provide a significant performance gain. On the other hand, using blockwise transfer, the integrity protection provided covers only the individual blocks, not the entire request or response. Thus, one or several of the block transfer would carry a Message Authentication Code (MAC) or signature that covers the entire request or response [21].

*a. JSON Web Signature*

Considering secure messaging in constrained environments, a new CoAP option (which is an object security approach), JSON Web Signature (JWS) [22] has been introduced to integrity protect individual request and responses. The JWS option contains a digital signature or MAC of the CoAP message using JavaScript Object Notation (JSON) based data structures. The validity of the signature/MAC is first checked by the endpoints supporting this scheme before accepting a message as a valid one [21].

*b. Sequence Number*

Another issue is replay attack. Even if a MAC/signature of the received message is valid, the message can still be old, or being replayed. Thus, the message sequence numbers can be used to protect from replay attack and verify freshness of responses. Using this approach, a CoAP client supporting the JWS shall store one sequence number per key it uses to better protect the integrity of a message. Similarly, a CoAP server supporting the JWS option shall store on sequence number per key for verification purpose [21].

*c. Access Control Server (ACS)*

A two-way authentication handshake has been proposed for handling the data exchange between two communication parties; client (subscriber) and server (publisher). In the Internet, the identities are typically established via PKC and identifiers are provided through X.509 certificates [23]. An X.509 certificate includes the public key of an entity, its common name and validity period, and signed by a trusted third party known as Certificate Authority (CA). The CA signature allows the receiver to detect modifications to the certificate, as well as to state that the CA has verified the identity of the entity that requested the certificate [24].

For the key establishment purpose, there should also be a trusted third party which can support client and server. The Authorization Server is responsible for authenticating the client on behalf of the server, or providing cryptographic keys or credentials to the client and/or server to secure the request/response procedure [21].

CoAP has no mechanism for authorization. To support end-to-end communication security for IoT, proper authentication of data publishing devices and access control throughout the network is required. An Access Control Server (ACS); a trusted entity and resource-rich server can be integrated in the system architecture. The access rights for the publisher (i.e. sensor nodes) of the network are stored in the ACS. Any subscriber who wants to initialize a connection with the publisher has to obtain an access ticket from the ACS. It is the responsibility of the ACS to verify that the subscriber has the right to access the information from the publisher. The publisher only has to evaluate the identity of the subscriber and has to verify the ticket received from the ACS. For this purpose, a unique identity is required for a publisher in the network [24].

*d. One-time Authorization Grant*

Another simpler approach to grant secure temporary access request is one-time authorization grant which is based on some freshness maintained between the ACS and publisher such as nonces or sequence numbers. ACS may keep a counter for each publisher, step the counter each time it generates new authorization and include the counter in the authorization information. Publisher accepts the fresh authorization information by comparing the counter value to the highest previously received counter [25].

*e. Delegated CoAP Authentication and Authorization Framework (DCAF)*

As constrained devices have limited system resources such as memory, stable storage and transmission capacity, it is difficult to realize authentication mechanism. Delegated CoAP Authentication and Authorization Framework (DCAF) has been proposed in [26] to help constrained devices with authorization-related tasks. Complex security tasks such as managing keys will be performed by the Authorization Managers (AM), which are less-constrained devices. A device that wants to access an item from another constrained device first has to gain permission in the form of a token from the node's AM. The main purposes of DCAF are to establish a DTLS channel with symmetric pre-shared keys (PSK) between a pair of nodes and to securely transmit authorization tickets. The communication is first initialized by a Client (who wants to access a resource) requesting an access ticket from the Client AM (CAM). The CAM after verifying the Resource Server (RS), will transmit the request to Server AM (SAM). If the Client is allowed to access the resource (according to the policies of the RS), the SAM generates a DTLS PSK for the communication between the Client and the RS, wrapped in an access ticket. The access ticket may have a lifetime as defined by the SAM. A time synchronization method is used to ensure that the ticket lifetime is interpreted correctly. When the lifetime ended, the RS should end the DTLS connection to the Client.

Consequently, accurate time measurement is important to determine the validity of certain security properties such as a public key certificate, access token or some other assertion [25]. Synchronization of security states with sleepy nodes is another complex issue to be tackled [3].

*C. Privacy Protection*

Regarding privacy issue, the intelligent data management should collect only the required data. The processing of the collected data has to be minimized according to a strict set of rules so that it cannot be re-used [27]. Authentication and access control are important to control who gets access to private data. Role-based access control systems should provide different accessibility levels for various types of

information depending on the requesting application [14]. Additionally, in case of emergencies, "opt-in" paradigms which enable the users to voluntarily express and confirm their awareness and willingness to share their personal data should be incorporated [27].

A privacy-preserving identification names should be able to reduce the possibility to link information from IoT devices (such as location, uptime and usage) with a specific person or group of people. To reduce the privacy challenges, a variety of techniques may be used such as (i) the usage of only aggregate information collected by the sensors, rather than exact information about individuals, and (ii) the usage of privacy-preservation mechanisms (e.g. k-anonymity, l-diversity and t-closeness) in reducing the accuracy of the data before sharing it with other entities. For example, faces in the videos can be blurred to reduce the likelihood of identification, and additional noise, spatial cloaking, and spatial delays can be introduced for mobile and location data [28]. The sub-domains of a DNS name related to location can also be encrypted by a shared key or public-and-private keys to prevent the disclosure of location [29].

Privacy of RFID data is possible by encrypting the code in a tag before transmission. However, such a solution may not be effective as it only protects the content of the tag, but not the unique identify at the tag. To solve this, dynamic encryption within the tag could be embedded, but comes at a cost because it requires the chip to perform computation. Such cost could be avoided by performing the cryptographic computations at the reader end and the resulting information will be stored in the tags. Such schemes commonly work with re-writable memory in the tags to increase security. The tags are encrypted and decrypted by the reader when they are sent to the server. The reader also has the capability to re-encrypt the tag with a different key and write it to its memory, so that the tag signal for an eavesdropper is different at different times. Repeated change in the encrypted representation of the tag prevents the eavesdropper from uniquely identifying the tag [28].

Another solution to make it difficult to read tags in an unauthorized way is to use the blocker tags. This approach exploits the collision properties of RFID transmission whereby when two RFID tags transmit distinct signals to a reader at the same time, a broadcast collision may occur to prevent the reader from deciphering either response. The blocker tags will only spam unauthorized users, thereby allowing the authorized readers to behave normally [28].

## V. Conclusions

This paper has presented the elements of security that need to be implemented in IoT and common security attacks. It is undeniable that security is important for designing a robust routing protocol. However, the implementation of security should be carefully tackled to avoid significant transmission overhead. It is hoped that the security aspects and approaches presented in this paper would be beneficial for further research in implementing secure communicating environment for IoT. Further, synchronization among nodes including sleepy nodes should be considered as it affects the validity of keys and access token exchanged.

## References


[1] Elkhodr, M., Shahrestani, S. and Cheung, H. 2013. The Internet of Things: Vision & Challenges. *IEEE 2013 Tencon - Spring*. 218–222.

[2] Kim, E., Kaspar, D, Gomez, C. and Bormann, C. 2012. RFC 6606 - Problem Statement and Requirements for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing. 1–32.

[3] Tsao, T., Alexander, R., Dohler, M., Daza, V., Lozano, A., and Richardson, M. 2015. RFC 7416 - Security Threat Analysis for ROLL RPL. 1–40.

[4] Tschofenig, H., Arkko, J., Thaler, D. and Mcpherson, D. 2015. Architectural Considerations in Smart Object Networking. draft-iab-smart-object-architecture-05.txt. 1–21.

[5] Kim, E., Kaspar, D., Gomez, C. and Bormann, C. 2012. Problem Statement and Requirements for 6LoWPAN Routing. draft-ietf-6lowpan-routing-requirements-10. 1–35.

[6] Garcia-Morchon, O., Kumar, S., Keoh, S., Hummen, R. and Struik, R. 2011. Security Considerations in the IP-based Internet of Things. draft-garcia-core-security-06. 1–45.

[7] Kushanlnagar, N., Montenegro, G., and Schumacher, C. 2007. RFC 4919 - IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. 1–12.

[8] Housley, R. 2004. RFC 3686 - Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP). 1–19.

[9] Montenegro, G., Kushalnagar, N., Hui, J. and Culler, D. 2007. RFC 4944 - Transmission of IPv6 Packets over IEEE 802.15.4 Networks. 1–30.

[10] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P. and Kivinen, T. 2014. RFC 7296 - Internet Key Exchange Protocol Version 2 (IKEv2). 1–142.

[11] Kim, E., Kaspar, D. and Vasseur, JP. 2012. RFC 6568 - Design and Application Spaces for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs). 1–28.

[12] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, J. and Alexander, R. 2012. RFC 6550 - RPL. 1–157.

[13] Park, S., Hamid, M.A. and Hong, C. S. 2005. Routing Security in Sensor Attack: HELLO Flood Attack and Defense. 1–9.

[14] Miorandi, D., Sicari, S., De Pellegrini, F. and Chlamtac, I. 2012. Internet of things: Vision, applications and research challenges. Ad Hoc Networks 10 (2012). 1497 - 1516.

[15] Seitz, L., Gerdes, S., Selander, G., Mani, M. and Kumar, S. 2015. ACE use cases. draft-ietf-ace-usecases-02. 1 - 24.

[16] Piro, G., Boggia, G. and Grieco, L. A. 2015. Layer-2 security aspects for the IEEE 802.15.4e MAC. draft-piro-6tisch-security-issues-03. 1–27.

[17] Struik, R., Ohba, Y. and Das, S. 2015. 6TiSCH Security Architectural Elements, Desired Protocol Properties, and Framework. draft-struik-6tisch-security-architecture-elements-01. 1–8.

[18] Richardson, M. 2015. 6tisch secure join using 6top. draft-richardson-6tisch--security-6top-04. 1–22.

[19] Shelby, Z., Hartke, K. and Bormann, C. 2014. RFC 7252 - The Constrained Application Protocol. 1–21.

[20] Rescorla, E. and Modadugu, N. 2012. RFC 6347 - Datagram Transport Layer Security Version 1.2. 1–32.

[21] Selander, G., Mattsson, J. and Seitz, L. 2015. Object Security for ACE draft-selander-ace-object-security-00. 1–26.

[22] Jones, M., Bradley, J. and Sakimura, N. 2015. RFC 7515 - JSON Web Signature. 1–59.

[23] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R. and Polk, W. 2008. RFC 5280 - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List CRL Profile.

[24] Schmitt, C. and Stiller, B. 2015. Two-way Authentication for IoT. draft-schmitt-ace-twowayauth-for-iot-01. 1–19.

[25] Seitz, L. and Selander, G. 2015. Problem Description for Authorization in Constrained Environments. draft-seitz-ace-problem-description-02. 1–19.

[26] Gerdes, S., Bergmann, O. and Bormann, C. 2015. Delegated CoAP Authentication and Authorization Framework (DCAF) draft-gerdes-ace-dcaf-authorize-01. 1–42.

[27] Vermesan, O. and Friess, P. 2010. Internet of Things - Converging Technologies for Smart Environments and Integrated Ecosystems. River Publishers Series in Communications.

[28] Aggarwal, A. A., Ashish, N. and Sheth, A. Chapter 12 - The Internet of Things: A Survey from the Data Centric Perspective. 383 - 428.

[29] Jeong, J. and Park, J. 2015. DNS Name Autoconfiguration for Internet of Things Devices. draft-jeong-homenet-device-name-autoconf-03.