# State of the art of IETF security related protocols for IoT

Renzo Efrain Navas, Laurent Toutain, Kumaran Vijayasankar

# State of the art of IETF security related protocols for IoT (*categorie: generale*)

Renzo Navas[1], Laurent Toutain, and Kumaran Vijayasankar[2]

[1] Telecom Bretagne, Department of Network, Security and Multimedia, Cesson-Sevigne, France
firstname.lastname@telecom-bretagne.eu
[2] Texas Instruments, WCS R&D, Dallas TX, USA
kumaran@ti.com

**Abstract.** The Internet of Things (IoT) is becoming a reality and the Internet Engineering Task Force (IETF) is the main open standardization body responsible for it. The IoT implies billions of new devices connected to the Internet and, while several problems like interoperability and routing have been solved, security solutions suited for IoT are still an active field of research. This document is a survey of the state of the art at IETF of security related protocols for IoT. The needed IETF background and a highlight of current efforts on security for IoT is offered. An insight of unsolved problems and future perspectives on IETF concludes this survey. This is an informational document and detailed description of the protocols is not on scope.

**Keywords:** ietf, iot, security, protocols, standarization, survey

## 1 Introduction

This document gives an overview of the current state of the art of the Internet Engineering Task Force (IETF) working groups dealing with several aspects of *security* on constrained nodes and networks. The *security* aspects addressed by the presented protocols include: encryption (*confidentiality*), message authentication (*integrity*), entity authentication, and fine-grained authorization.

Security-related protocols and architectures were historically aimed for traditional networks -like the Internet- and devices. They where not conceived to work on constrained nodes or networks; these constraints include for example: nodes with limited RAM, ROM, CPU power and energy, high latency and unreliable network transmissions.

The aforementioned constraints add additional challenges when implementing or designing security protocols and architectures. Most of them simply cannot run on constrained environments. Hence new protocols and architectures need to be defined for achieving security goals on constrained nodes and networks.

The aim of the IETF is to re-use to the greatest extent possible the already defined security protocols and architectures adapting them to the constrained world of the Internet Of Things (IoT).

The rest of this paper is structured as follows: Chapter 2 presents a chronological summary of the IETF IoT-oriented working groups and most important protocols. Chapter 3 presents the fundamental standards that are the base for the security-related protocols. Chapter 4 deals specifically with the IETF security-related protocols categorizing them according to the layer they apply: network, transport or application. Chapter 5 presents the Authentication and Authorization Framework for IoT: *ACE*, currently being defined at IETF. Finally, Chapter 6 offers a summary of the unsolved security problems that yet need to be addressed and future axes of research at IETF.

## 2 A brief history of the IETF IoT-oriented working groups

The first efforts of IETF protocols adapted for the Internet of Things (IoT) date back to 2005 with the 6LoWPAN (IPv6 over Low power WPAN) working group (WG) which provided the first adaptation of IPv6 for constrained-node networks on RFC4944: *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*[1] published on 2007-09. The ROLL (Routing Over Low power and Lossy networks) WG addressed the routing problem on *6LoWPANs* with the design and publication of *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks (LLN)*[2] published on 2012-03. These documents provided the foundations IoT world allowing IPv6 connectivity on Low-Power and Lossy Networks.

Since around the year 2010 with the start of the CoRE (Constrained RESTful Environments) WG we can observe a growing interest on the IETF about constrained environment related working groups and protocols. The CoRE WG deals with application-level goals and one of his most valuable outputs has been *The Constrained Application Protocol (CoAP)*[3] (RFC 7252). The 6lo (IPv6 over Networks of Resource-constrained Nodes) WG further continues the work of the 6LoWPAN WG and aims to facilitate IPv6 connectivity over constrained node networks with heterogeneous link layer technologies; similarly, 6TiSCH (IPv6 over the TSCH -Timeslotted Channel Hopping- mode of IEEE 802.15.4e) WG focus on enabling IPv6 over the TSCH mode of the IEEE 802.15.4e standard. LWIG (Light-Weight Implementation Guidance) WG publishes guidelines for implementation of all these protocols in constrained nodes and has standardized a common terminology to be used on future standards.

The IPv6 adaptation 6LoWPAN, the RPL routing protocol, and the CoAP application protocol have become the pillar protocols for connectivity and interoperability on the IoT. Other IoT-related protocols are *The User Datagram Protocol (UDP)* published on 1980 and the *Datagram Transport Layer Security (DTLS)* protocol published on 2006, even if they pre-date the IoT paradigm they are well suited to solve the IoT constraints. All this protocols constitute what

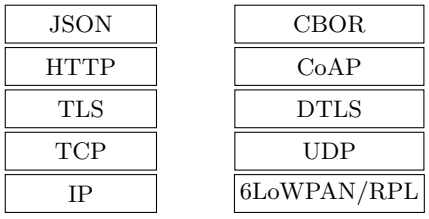


**Fig. 1.** IETF Protocols stack for: (a) Standard Internet; (b) Internet of Things;

has become the basic IoT stack. This IoT stack[3] side by side with the standard Internet protocols stack can be seen on Figure 1.

**Security-oriented** WGs started to arise starting from the year 2013 with the **DICE (DTLS In Constrained Environments) WG** profiling DTLS for IoT and later with the **ACE WG** dealing with Authentication and Authorization on Constrained Environments. The 6lo (IPv6 over Networks of Resource-constrained Nodes) and 6TiSCH WGs also started to focus on security-related milestones. In general all the IoT groups started to focus on security once the deployment of the base IoT protocols started to become a reality but the basic security needs where not yet addressed.

## 3 Fundamentals

This chapter presents the fundamental standards that are the base for the security-related protocols that will be presented later.

### 3.1 Nomenclature

*Terminology for Constrained-Node Networks*[4] (RFC7228) defines a common terminology that has been used thoroughly on all the other IETF constrained environments new protocols. One important output is the distinction of three different *Classes of Constrained Devices* shown on Table 1.

**Table 1.** Classes of devices according to RFC7228 [4]

| **Device Class** | **RAM (KB)** | **Flash (KB)** |
|---|---|---|
| Class 0 | << 10 | << 100 |
| Class 1 | ≈ 10 | ≈100 |
| Class 2 | 100 | 250 |

[3] With the addition of the *CBOR* binary data format, that will be introduced further on this paper.

Class 0 devices are assumed to not have the resources required to communicate directly with the Internet in a secure manner. Class 2 devices have not difficulties implementing standard protocols. Class 1 devices are assumed on most new security solutions and protocol adaptations; in other words, current IETF for IoT security proposals are targeted to devices with 10 KB of RAM 10KB and 100 KB of Flash.

Having a standardized terminology helps to properly describe the different problematics on IoT in a consistent way and propose solutions accordingly.

### 3.2 CoAP Protocol and Security

The Constrained Application Protocol (CoAP)[3] (RFC 7252) is a specialized web transfer protocol for use with constrained nodes and constrained networks. One of its main goals is to provide a RESTful transfer service similar to HTTP, but simplified for the use on constrained devices on constrained networks. CoAP provides a request/response interaction model between application endpoints, supports built-in discovery of services and resources, and includes key concepts of the Web such as URIs and Internet media types. Even if CoAP has been designed to run over UDP, it can be adapted to run on top of any other datagram oriented protocol (e.g.: CoAP-over-SMS, CoAP-over-LoRaWAN, CoAP-over-Bluetooth Low Energy).

**Security in CoAP**. The standardized approach for securing CoAP (*securing* meaning *confidentiality* and *integrity*) is using the *Datagram Transport Layer Security (DTLS) Version 1.2* (RFC 6347). CoAP assumes that the cryptographic credentials are already previsioned, and defines four security modes:

1. **NoSec**: There is no protocol-level security (DTLS is disabled).
2. **PreSharedKey**: DTLS is enabled and pre-shared keys ciphersuites are used.
3. **RawPublicKey**: DTLS is enabled and the device has an asymmetric key pair without a certificate that is validated using an out-of-band mechanism.
4. **Certificate**: DTLS is enabled and the device has an asymmetric key pair with an X.509 certificate.

CoAP does not have primitives to assure neither authentication or authorization; if these security services are required they will have to be provided either by *communication security* (i.e., IPsec or DTLS) or by *object security* (within the payload). The DICE (DTLS In Constrained Environments) and ACE (Authentication and Authorization for Constrained Environments) working groups were created to asses the remaining challenges and complete an appropriate framework needed for a secure IoT environment.

### 3.3 CBOR: Concise Binary Object Representation

The *Concise Binary Object Representation* (CBOR)[5] (RFC 7049) is a binary data format inspired by JSON and aimed at compact representation of most common data types used at Internet standards; it also has the explicit goals of

a lightweight implementation in terms of code and ram needed, and no needed schema description to decode. It is normally used on top of CoAP to represent information.

## 4 IETF security at different layers

IETF has developed IoT security solutions at all the layers it is competent, this includes network-layer (Layer 3) and above. Security at link-layer (Layer 2) and below are not on scope of IETF. Different types of applications will need security at different layers. There is a rough consensus currently at IETF to go for application layer security (also called *object security*), on spite of the others: network-layer solution is seen as cumbersome, difficult to implement and not so lightweight; transport-layer provides some end-to-end (e2e) security problems at the presence of proxies; on the contrary application layer is seen as the most flexible solution and suitable for caching and proxying while maintaining e2e security properties. We will present security solutions at each layer.

### 4.1 Security at network-layer

Security at network-layer is the less active field at IETF. Security is based on adaptations of the IPsec protocol suite: IKEv2 and ESP. The LWIG working group is the home for this efforts. RFC 7815: *Minimal Internet Key Exchange Version 2 (IKEv2) Initiator Implementation* [6] is the most important document and presents a lightweight version of the IKEv2 protocol; this protocol is used for performing mutual authentication and establishing and maintaining security associations (fresh keys) on the IPsec suite. To encrypt the data a minimal version of the IP Encapsulation Security Payload (ESP) is being currently worked on the same working group, however the future of IPsec encryption for IoT is not clear, and seems to be relegated to higher layer security solutions.

### 4.2 Security at transport-layer

Transport-layer solution for IoT is provided by Datagram Transport Layer Security (DTLS). A special working group *DTLS In Constrained Environments (dice)* was created to define a DTLS profile suited for IoT. The output document is *Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things* (RFC 7925 [7]). This document dictates how to set available configurable options and protocol extensions: e.g. setting timers values, choosing appropriate DTLS ciphersuites. The three types of credentials to use correspond which the mandate of CoAP: (1) for PSK the mandatory ciphersuite is `TLSPSK_WITH_AES_128_CCM_8`; (2) for raw public key the ciphersuite is `TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8` which uses elliptic curve cryptography; (3) for certificates the mandatory ciphersuite is the same as raw public key, but the key will be wrapped on a X.509 v3 certificate. After the DTLS authentication credentials have been used on the DTLS handshake protocol all

methods encrypt with the same symmetric algorithm AES-128-CCM, with an 8 byte tag/integrity value. Once we encrypt the DTLS per datagram overhead is of 13 bytes without counting the tag.

One of the advantages of transport-layer security is that can be used to transport any application data, including CoAP. Hence most IETF IoT protocols assumed at least a PSK between nodes, the establishment of a DTLS channel following the guidelines of DICE and then security was guaranteed. However, one of the drawbacks of this solution is that end-to-end security cannot be achieved in the presence of CoAP proxies (forwarding, caching), the DTLS secure channel will have to be terminated at the proxy, this led to the definition of higher layer security solutions.

### 4.3 Security at application-layer

Security at this layer, also referred as *object security*, has the advantage that can be maintained end-to-end and the security properties can be set on a per-message basis. The pillar of object security is *CBOR Object Signing and Encryption (COSE)* [8]. COSE describes how to create and process encryption, signatures and message authentication codes using CBOR for serialization.COSE will has more flexible security properties than DTLS but at the cost of more overhead. COSE is used to build upon other solutions like *Object Security of CoAP (OSCOAP)* [9]. OSCOAP provides end-to-end encryption, integrity, and replay protection of CoAP messages. It has two modes one only protecting payload, and other also protecting CoAP certain options and header fields. It also provides a secure binding between CoAP request and response messages, and freshness of requests and responses.

### 4.4 Security at layers summary

A summary of the mentioned security solutions contrasted with the IoT non-secured protocols and standard Internet protocols can be seen on Table 2.

**Table 2.** IETF security at different layers: Protocols

| Layer | Std. Internet | Secure Std. Internet | IoT | Secure IoT |
|---|---|---|---|---|
| **Application** | HTTP | HTTPS | CoAP,CBOR | OSCOAP,COSE |
| **Transport** | TCP | TLS | UDP | DTLS |
| **Network** | IP | IKEv2+IPsec(ESP) | 6LoWPAN | Min.IKEv2+ESP |

## 5 An Authentication and Authorization Framework: OAuth 2.0 for IoT

The Authentication and Authorization for Constrained Environments (ACE) working group is the most active on developing a comprehensive security so-

lution for the IoT. As his title depicts the main objectives are *Authentication* and *Authorization*, *encryption* will be leveraged on solutions previously mentioned like iot-profiled-DTLS or object security. The working group has two explicit goals: (1) Produce use cases and requirements and (2) Identify authentication and authorization mechanisms suitable for resource access in constrained environments. The first goal has been addressed and the second goal is being addressed by the work-in-progress: *Authentication and Authorization for Constrained Environments (ACE)* [10]. This section will offer an overview of the current ACE solution. The building blocks of the solution are: CoAP, CBOR/COSE, the OAuth 2.0 framework and the proof-of-possession tokens.

### 5.1 Standard OAuth 2.0

OAuth 2.0 is an open standard that enables a resource owner to grant a third-party limited access to a protected resource; is designed to be used over HTTP and TLS. *Access Tokens* are authorization credentials that grant access to protected resources. The *Proof-of-Possession (PoP) Tokens* are access tokens bound to a specific cryptographic key. To access a protected resource not only the token must be possessed but also the possession of the associated key must be proven. One important security property of PoP tokens is that they can be **transported over unsecured channels**.

### 5.2 ACE's IoT OAuth 2.0

ACE's OAuth v2.0 for the IoT uses PoP access tokens; HTTP is replaced by CoAP and JSON by CBOR, CoAP runs over UDP instead of TCP, and hence DTLS should be used instead of TLS. Application-layer security solutions are also envisioned and are based on COSE (CBOR Encoded Message Syntax)[8] and OSCOAP (Object Security of CoAP).

**OAuth Entities**: The main entities involved in the OAuth framework are:

- *Resource Server (RS)*: An entity which hosts a protected resource.
- *Client (C)*: An entity which attempts to access a protected resource on a RS.
- *Authorization Server (AS)*: An entity that enforces the *Resource Owner*'s policies, prepares and endorses authorization and authentication data.

**OAuth Message Flows**: The procedure that allows C to get access to a protected resource is the following:

1. C sends a *Token Request* message to the AS.
2. If C is authorized AS generates and sends to C the *Access Token* (opaque to C) and *Client Information* (e.g. contains the key bound to the PoP access token).
3. C sends the *Access Token* to RS and the protected resource Request
4. RS validates the request with the associated access token, if successful, responds with an (encrypted) representation of the protected resource.

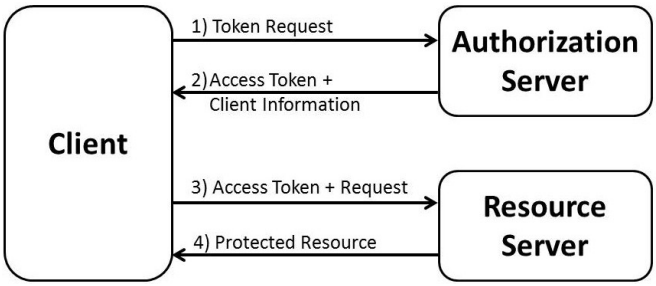


**Fig. 2.** OAuth 2.0 Basic Messages Flow and Entities

Figure 2 represents the basic OAuth 2.0 message flow and its main entities.

The PoP Token offers **the secure establishment of an authenticated key -with associated authorization permissions- between previously unknown parties and over an unsecured channel**. This fresh key can be further used to establish authenticated and secured communications between these parties.

## 6 IETF perspectives, industrial applications and conclusion

*IETF Perspectives.* The IETF is providing tools that permit today to offer an IoT-suited security solution for most of the devices and use cases. There is still unsolved problems. Provisioning-bootstrapping of cryptographic material is also being studied at the Thing-to-Thing research group, all current solutions assume some pre-previsioning or a trusted third party. Cryptographic generated Identities, ID-based cryptography and new elliptic curves are also current topics studied at the crypto forum research group. Solutions for Low-Power Wide Area Networks with ultra-low speeds/bandwidth are needed and will be addressed on a future WG: *LPWAN (IPv6 over Low Power Wide-Area Networks).*

The ACE WG is defining a comprehensive security framework that aims at solving most of IoT use cases, but time awareness is required at the constrained node. New types of time-constrained terminology are needed and solutions that do not rely on time synchronization will need to be proposed.

*Industrial.* Industrial applications are using IETF's already well-established IoT protocols such as CoAP adapting it to their needs (e.g.: running on TCP). Most recent protocols (e.g: COSE) or in-progress ones (ACE Framework) will take time to be adopted on the industrial world due to the conservative nature of it. However, we believe that the time required for the IETF protocols to be adopted by the industry will likely decrease. We base our last assertion on the good interaction between IEEE (a body more associated with the industry) and IETF for example defining cross-layer solutions at the 6TiSCH working group;

also the now forming LPWAN WG has some key technology players (SIGFOX, LoRA Alliance, Wi-SUN, NarrowBand-IOT) being involved, and ACE WG has some industrial referents such as Ericsson and ARM; meaning they will give valuable input to future protocols, making them more aligned with the industrial needs, and more likely to be quickly adopted.

*Conclusion.* Secure Identity representation, Bootstrapping, Privacy, and Multicast-security are some of the yet unsolved problems. IETF might never arrive to a one-fits-all solution, but the open nature of the protocols and layered design are enough tools to make security for IoT a reality.

## References


1. G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," RFC 4944 (Proposed Standard), Internet Engineering Task Force, Sep. 2007, updated by RFCs 6282, 6775. [Online]. Available: http://www.ietf.org/rfc/rfc4944.txt
2. T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550 (Proposed Standard), Internet Engineering Task Force, Mar. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6550.txt
3. Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," RFC 7252 (Proposed Standard), Internet Engineering Task Force, Jun. 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7252.txt
4. C. Bormann, M. Ersue, and A. Keranen, "Terminology for Constrained-Node Networks," RFC 7228 (Informational), Internet Engineering Task Force, May 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7228.txt
5. C. Bormann and P. Hoffman, "Concise Binary Object Representation (CBOR)," RFC 7049 (Proposed Standard), Internet Engineering Task Force, Oct. 2013. [Online]. Available: http://www.ietf.org/rfc/rfc7049.txt
6. T. Kivinen, "Minimal Internet Key Exchange Version 2 (IKEv2) Initiator Implementation," RFC 7815 (Informational), Internet Engineering Task Force, Mar. 2016. [Online]. Available: http://www.ietf.org/rfc/rfc7815.txt
7. T. Fossati and H. Tschofenig, "Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things," RFC 7925, Jul. 2016. [Online]. Available: https://rfc-editor.org/rfc/rfc7925.txt
8. J. Schaad, "CBOR Object Signing and Encryption (COSE)," Internet Engineering Task Force, Internet-Draft draft-ietf-cose-msg-18, Sep. 2016, work in Progress. [Online]. Available: https://tools.ietf.org/html/draft-ietf-cose-msg-18
9. G. Selander, J. Mattsson, L. Seitz, and F. Palombini, "Object Security of CoAP (OSCOAP)," Internet Engineering Task Force, Internet-Draft draft-selander-ace-object-security-05, Jul. 2016, work in Progress. [Online]. Available: https://tools.ietf.org/html/draft-selander-ace-object-security-05
10. E. Wahlstroem *et al.*, "Authentication and Authorization for Constrained Environments (ACE)," IETF, Internet-Draft draft-ietf-ace-oauth-authz-02, Jun. 2016, work in Progress.