WHITE PAPER

OPEN CONNECTIVITY FOUNDATION®

Overview: Open Connectivity Security Specification

OPEN CONNECTIVITY FOUNDATION MEMBER CONFIDENTIAL

Abstract:

While IoT security remains a largely discussed industry concern, it is possible to properly secure devices with the right protocols and cybersecurity measures in place, such as Open Connectivity Foundation's security solution. Any IoT security solution must be holistic in its approach, implementing security measures across multiple layers that include hardware, network, and software security. Open Connectivity Foundation's focus on the software application layer is a crucial first step in the IoT design process.

The OCF Core Framework, created with the expertise of OCF's extensive membership base, provides a complete foundation for an IoT solution. Compliant with most of the prominent security baseline guidelines, this framework ensures that security is not an afterthought – and is instead a fundamental part of the development process.

OCF experts will discuss the current security landscape within the IoT and how the OCF Core Framework offers the best in class security, using the latest technologies and proven standards.

Table of Contents

The Four Legs of Secure and Interoperable IoT

Many things both in nature and man-made have four legs. Four legs provide a solid foundation and balance over the center of gravity. Building a holistic solution for secure IoT that is interoperable between various vendors and manufacturers also requires four things for balance and completeness:

(1) You must have a well-defined, clear specification with a common resource model created and vetted by industry experts.
(2) You must have a rigorous testing and certification strategy that is global in scope.
(3) You must have a reference architecture implemented in code that matches the specification, conforms to certification, and provides an example for implementation.
(4) You must have an innovative and secure cloud architecture that is vendor agnostic.

The Open Connectivity Foundation (OCF) brings these four things: specification, certification, implementation, and cloud innovation; together with an industry-leading, security-first design approach. This makes the OCF more than a protocol, and greater than an ecosystem. The OCF is truly an IoT-Biome where over 500 companies have come together over the last five years and built a framework for secure interoperability of IoT devices.

Overview of OCF Architecture

Protocol Stack

OCF runs on top of IPv6, and leverages open standards to provide message transport, session, encryption, and discovery. OCF is agnostic to the underlying layer 2 connection and can run over Wi-Fi, Ethernet, and Thread. The full stack used by OCF is shown in Figure 1.

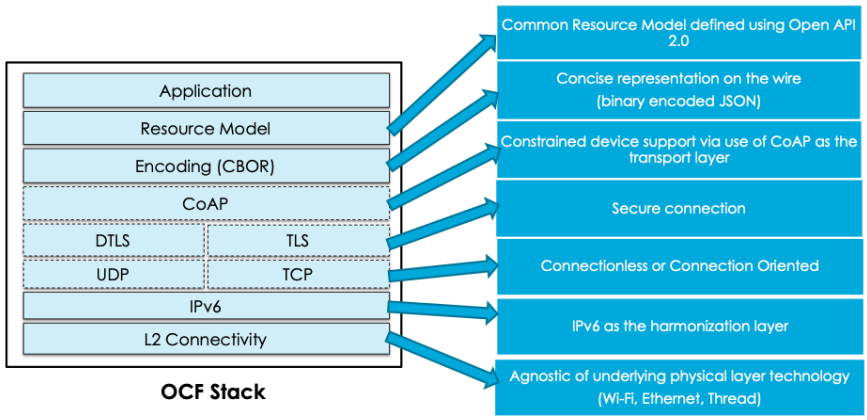

Figure 1: OCF Protocol Stack

Client-Server Architecture

OCF uses the common client-server architecture where each device can act as both a client (requestor of information or invoker of actions) and a server (supplier of information or realization of actions). Servers have resources (static information or information that can be updated or actuated) that can be requested by clients. For example, a smartphone acting as a client can send a command or request information from a lightbulb acting as a server. OCF uses the constrained application protocol (CoAP), defined in RFC 7252, as the transport layer, CoAP is designed for resource-constrained internet devices.

All communication between clients and servers is accomplished using a REpresentational State Transfer (REST) approach where the client requests information from, or sends a command to, a particular resource, e.g. the state of the light true(on) or false(off). All operations are done using Create, Retrieve, Update, Delete, Notify (CRUDN) conventions to interact with resources. All requests payloads are defined by OCF specifications using OpenAPI2.0, and transferred over the wire using CBOR encoding to reduce the size of the message payloads.

The server verifies the client's identity and checks to see if it has the correct permissions to access the resource. If everything checks out, then the server responds with the requested information or performs the requested action. Figure 2 shows how this works with the lightbulb example. Here the resources are properties of the light bulb, such as its state (on/off/dimming) or actions that can be performed on the bulb (power[true/false], change dimming level, etc.).

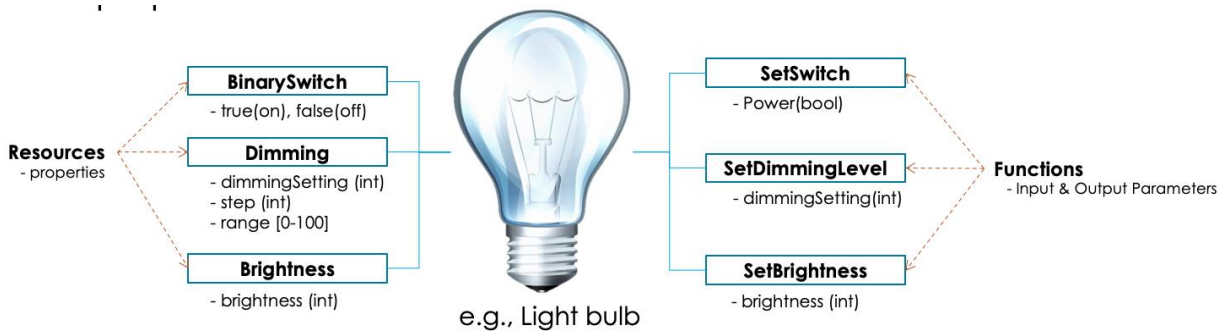

Figure 2 Lightbulb as a server

Overview of OCF Security

The Life Cycle of a Secure IoT Device

Making sure your device will securely work with other devices in your home begins before you even take it out of the box. Certified OCF devices will have the OCF logo on the box, letting you know that this device has passed the rigorous compliance testing required by each OCF-Certified device. To pass certification, each device must pass a comprehensive, automated, test suite that proves it will work with other OCF certified devices and that it meets the specification's normative security requirements.

The life cycle of an OCF device has two primary stages, unowned and owned, within these stages are various operation states that dictate what and when the device is allowed to do. An unowned device is like a device that has just been unboxed. To add it to your smart-home you must take ownership of the device i.e. bring it on to your network, and give it credentials and permissions so that it can interact with your other devices. This step is also referred to as onboarding. The complete set of operational states of an OCF device are listed below:

1. **Hard Reset (Factory Reset)**

   When you bring home a new device take it out of the box and plug it in the device starts in a clean-slate mode where all settings are the manufacturer defaults.

2. **RFOTM (Pairing Mode)**

   Once the device boots up it goes into a pairing mode, called "Ready For Ownership Transfer Method" (RFOTM). This pairing mode allows you to scan the network and see the new device. Once you find the new device you can select it and set it up. This setup configuration is done with a configuration tool called the Onboarding Tool (OBT) which is the primary piece of software that is used to configure a device on an OCF network; it is expected that an OBT is a resident part of an application that could be resident on a cell-phone, tablet, or similar. The OCF specification defines three algorithmic methods for transferring ownership:

   a. Just Works: Encrypted, not authenticated, based on Anonymous Diffie-Hellman
   b. Random PIN: Encrypted, mutually authentication between device and OBT. This requires that the device have a way to display the PIN (12 digits, or 8-character alphanumeric, or 7-character case-sensitive alphanumeric)
   c. Manufacturer Certificate: Encrypted, Public Key Infrastructure (PKI) based. This is the only method that provides strong cryptographic assurance of the device identity (type, model, manufacturer).

3. **RFPRO (Configuration)**
   After the device has been selected, it then needs to be configured. This configuration is done in the Ready for Provisioning (RFPRO) state. This is the state where credentials are configured, and access control lists are defined. The device can return to this state if additional configuration is needed.

4. **RFNOP (Normal Operation)**
   After the device is configured, then the device transitions to the Ready for Normal Operation (RFNOP) state. The device will spend the majority of its time in this state.

5. **Hard Reset (When Decommissioning)**
   When the device has reached end-of-life, or ownership will be transferred, the device should be hard reset. This erases all of the configuration on the device, including Wi-Fi and operational credentials, roles, routines (scenes), etc. and protects users from both security and privacy breaches.

6. **Transfer or Recycle**
   The device is now in a factory default state where it can be re-owned and incorporated into a new smart-home environment or be recycled.

Figure 3 shows the OCF lifecycle stages and operational states.

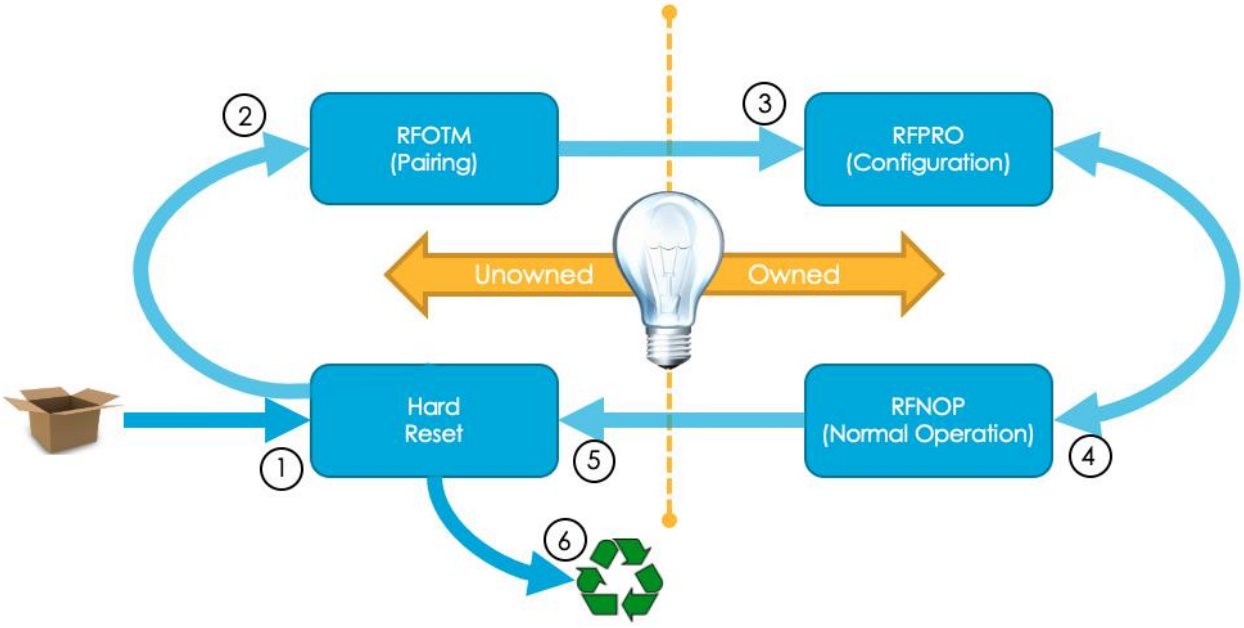

Figure 3: Device Lifecycle

Cybersecurity Fundamentals

The OCF takes the best practices of cybersecurity and applies them to IoT devices. This requires applying the six core security concepts of the A.I.C triad (Availability, Integrity, Confidentiality) and the A.A.A triple (Authorization, Access Control, and Auditing).

A.I.C Triad

Availability

Are the devices available when they need to be, i.e. does the light come on when you need it?

OCF devices can support cloud connectivity, but they MUST support local network operations. The availability of devices becomes critically important in several smart home systems such as security, HVAC, smoke, and flood detectors, even if the wide area network or cloud service is not available.

Integrity

Protect data from modification, making sure that the commands sent to a device cannot be corrupted. In addition to confidentiality, OCF uses the underlying protocol (TLS/DTLS) to provide a hashing algorithm for integrity protection, such as the Secure Hash Algorithm (SHA) in combination with Keyed-Hashing for Message Authentication (HMAC). These protocols make sure that messages between devices cannot be changed by unauthorized entities.

Confidentiality

Communication, both device-to-device and device-to-cloud, is encrypted using transport layer security (TLS) or its datagram equivalent, DTLS. This protects against eavesdropping, tampering, and message replay attacks. The encryption in OCF follows NIST SP 800-57 approved algorithms and key lengths. OCF encryption can be done using either symmetric keys provisioned to each device or using asymmetric keys with a common installed root of trust.

A.A.A Triple

In addition to the AIC triad the A.A.A triple defines how access is authorized, controlled, and recorded. The authentication and authorization process in OCF is comprehensive and takes the form of access control lists (ACLs). OCF employs the concept of least privilege. This means that access to any resource by any client must be explicitly granted.

Authentication

The mechanism of verifying the incoming request against an identifying credential.

OCF provides two identities:

1. Manufacturer Certificate Credential – this must be immutable, unique, and attestable. It is used for onboarding the device on to the network and comes in the form of a certificate installed on the device at the factory and is backed by a PKI.

2. Operational Credential – unique and assigned by the network or network administrator.

When an OCF device is onboarded, it is assigned a unique operational credential. The device uses this credential to authenticate its identity to other devices in the OCF network.

Authorization

Authorization is the permission or privilege to access, update or actuate a resource. In OCF this authorization takes the form of an Access Control Entry (ACE) which is a set of permissions, operations, resources, and identities. If there is a matching ACE, then access to the resource or action is permitted; otherwise access is denied. Many ACEs can be grouped together into a list called an access control list (ACL).

Auditing

The third leg of the AAA framework involves tracking and logging when the security context changes. This could be when a credential is added or deleted, or when an access is granted or denied. Logging is needed both for forensic analysis, and for real-time understanding of system failures. When something goes wrong, it is important to understand what chain of events led to a failure, and what devices are impacted. In OCF audit logging can be done either locally on the device or remotely.

Subject-based Access Control (SBAC)

Subject-based access control is where the identity presented by the client matches the identity stored in an ACE for a particular resource.

Role-based Access Control (RBAC)

To provide a solution that provides more flexibility and scalability than simple subject-based access control, OCF employs a role-based access control (RBAC) method for controlling access to resources. IN RBAC, the permissions to perform certain operations on resources are assigned to specific roles. A role credential is then configured on the client. Any client with that role credential can then perform those actions on the server's resources. In OCF role credentials are always in the form of certificates.

Secure Resources

In addition to the resources that are shown above in Figure 2, OCF defines several resources that are designated as special security-related resources. The specification refers to these resources as Secure Virtual Resources (SVRs). The SVRs store things like credentials, and the configuration state of the device. Because these resources contain critical information, access to these resources must be over a mutually authenticated and encrypted connection and can be changed only when the device is in configuration mode or (RFPRO). These SVRs are changed only by an appropriately privileged client, typically an OBT.

The following are four primary groups of SVRs:

- Device Ownership Transfer Resource (/oic/sec/doxm) – store and manage device ownership status

- Provisioning Resource (/oic/sec/pstat) store and manage Device Provisioning status
- Credential Resource (/oic/sec/cred) store and manage Device credentials
- Access Control List (/oic/sec/acl2) store and manage the Access Control Entry for the Resource Server

Credentials

OCF devices support multiple credential types:

- Pairwise Secret Key
- Asymmetric credentials:
    - Raw asymmetric key
    - Identity certificate w/ key pair – used to establish a secure session
    - Role certificate w/ key pair – used to assert a role within a session

Additional Online Resources:

- https://openconnectivity.org/developer/specifications

Resource Type Definitions

- Core Resources: https://github.com/openconnectivityfoundation/core
- Core Extension Resources: https://github.com/openconnectivityfoundation/core-extensions
- Bridging Resources: https://github.com/openconnectivityfoundation/bridging
- Cloud Resources: https://github.com/openconnectivityfoundation/cloud-services
- Security Resources: https://github.com/openconnectivityfoundation/security-models

Vertical Resources and Derived Models:
https://openconnectivityfoundation.github.io/devicemodels/docs/index.html

Overview of Baseline Comparison

Over the last few years government entities and industry organizations have developed guidelines and requirements documents that lay out baseline security requisites for IoT devices. These guidelines vary on scope and intent, however, taken as a whole these IoT security baselines present a set of common tools with which measure and provide external validation of IoT security models.

Five influential IoT baselines: NIST 8259, CTA-C2, ENISA, UK IoT Security Standards, and ETSI have been mapped to the OCF security architecture. This mapping gives an overarching picture of the completeness of the OCF security model, especially considering that every mandatory device requirement in OCF has a corresponding certification test, or if a test is not feasible then the manufacturer must attest that they meet the criteria established in the specification.

Three criteria were derived to determine if OCF meets each base line:

Met: There is one or more clauses in the OCF specification that meets the baseline requirement.

Unmet: Not Applicable: There is no clause in the OCF specification that meets the baseline requirement, *but* the requirement is out of scope for a device centric specification, or the requirement is implementation/vendor specific. Examples of out of scope include user data protections in the cloud being both out of scope for a device specification and vendor specific.

Unmet: Not Implemented: There is not a clause in the OCF specification that meets the baseline requirement, however the requirement is in scope.

Figure 4 shows the total applicable requirements OCF's security model meets for each baseline.

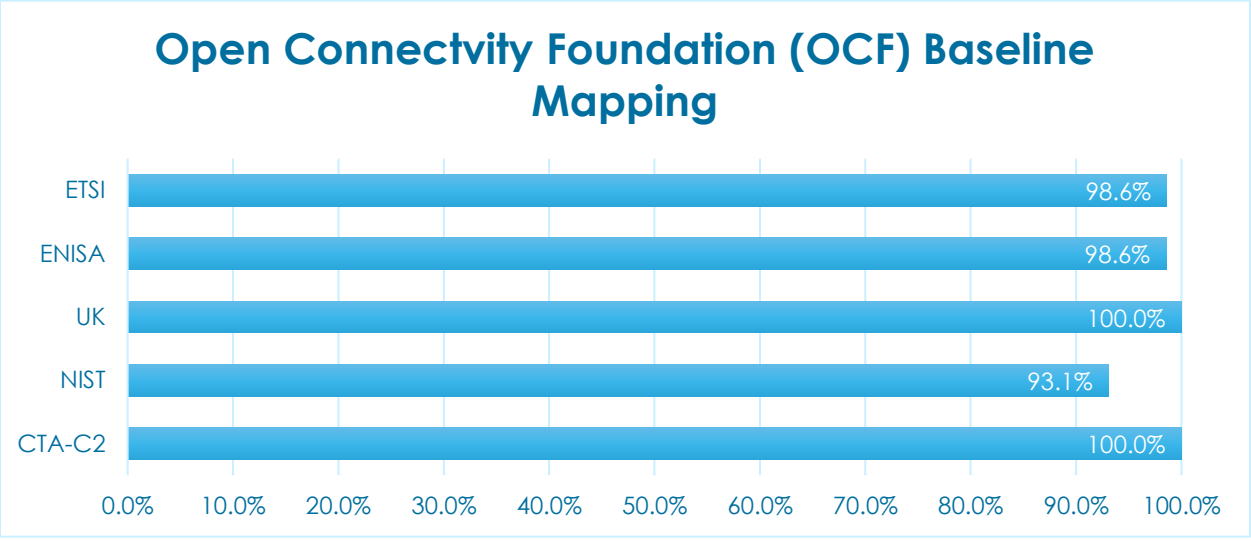

Figure 4 Baseline Measures

List of IoT Security Baselines

Baseline Overview webpage: https://openconnectivity.org/technology/ocf-security/

NIST 8259: National Institute of Standards and Technology NISTR 8259 consists of six main requirements with a total of 22 sub-requirements. NISTR 8259 is focused specifically on IoT devices and does not layout guidelines for service providers and cloud applications.

CTA-C2: The Consumer Technology Association's "Convene the Conveners" (CTA-C2) consensus on IoT device security is designed to bring together many vertical interests within the IoT market. It consists of ten baseline device capabilities, three baseline device lifecycle capabilities and several additional requirements to be phased in over time.

ENISA: European Union Agency for Cybersecurity' Baseline Security Recommendations for IoT consists of 15 main requirements with 57 sub-requirements.

UK: Code of Practice for Consumer IoT Security from the UK's Department for Digital, Culture, Media and Sport lays out 13 requirements for IoT devices.

ETSI: European Telecommunications Standards Institute has a Cyber Security for Consumer Internet of Things Baseline Requirements that lists 14 main requirements with 67 sub-requirements. ETSI also lists data protection provisions