1 LEGAL DISCLAIMER

NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS GRANTING YOU ANY KIND

OF LICENSE IN ITS CONTENT, EITHER EXPRESSLY OR IMPLIEDLY, OR TO ANY

INTELLECTUAL PROPERTY OWNED OR CONTROLLED BY ANY OF THE AUTHORS OR

DEVELOPERS OF THIS DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED

ON AN "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW,

THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY DISCLAIM ALL OTHER

WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT

COMMON LAW, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF

MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OPEN INTERCONNECT

CONSORTIUM, INC. FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-

INFRINGEMENT, ACCURACY OR LACK OF VIRUSES.

The OCF logo is a trademark of Open Connectivity Foundation, Inc. in the United States or other

countries. \*Other names and brands may be claimed as the property of others.

Copyright © 2017-2021 Open Connectivity Foundation, Inc. All rights reserved.

Copying or other form of reproduction and/or distribution of these works are strictly prohibited

415 Introduction

This document, and all the other parts associated with this document, were developed in response

to worldwide demand for smart home focused Internet of Things (IoT) devices, such as appliances,

door locks, security cameras, sensors, and actuators; these to be modelled and securely controlled,

locally and remotely, over an IP network.

While some inter-device communication existed, no universal language had been developed for

the IoT. Device makers instead had to choose between disparate frameworks, limiting their market

share, or developing across multiple ecosystems, increasing their costs. The burden then falls on

end users to determine whether the products they want are compatible with the ecosystem they

bought into, or find ways to integrate their devices into their network, and try to solve interoperability

issues on their own.

In addition to the smart home, IoT deployments in commercial environments are hampered by a

lack of security. This issue can be avoided by having a secure IoT communication framework, which

this standard solves.

The goal of these documents is then to connect the next 25 billion devices for the IoT, providing

secure and reliable device discovery and connectivity across multiple OSs and platforms. There

are multiple proposals and forums driving different approaches, but no single solution addresses

the majority of key requirements. This document and the associated parts enable industry

consolidation around a common, secure, interoperable approach.

1 Scope

This document defines security objectives, philosophy, Resources and mechanism that impacts

OCF base layers of [ISO/IEC 30118-1. ISO/IEC 30118-1](#_bookmark3) contains informative security content. The

OCF Security Specification contains security normative content and may contain informative

content related to the OCF base or other OCF documents.

2 Normative References

The following documents, in whole or in part, are normatively referenced in this document and are

indispensable for its application. For dated references, only the edition cited applies. For undated

references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 30118-1 Information technology -- Open Connectivity Foundation (OCF) Specification --

Part 1: Core specification

<https://www.iso.org/standard/53238.html>

Latest version available at:

<https://openconnectivity.org/specs/OCF_Core_Specification.pdf>

ISO/IEC 30118-3 Information technology -- Open Connectivity Foundation (OCF) Specification --

Part 3: Bridging specification

<https://www.iso.org/standard/74240.html>

Latest version available at:

https://openconnectivity.org/specs/OCF\_Bridging\_Specification.pdf

OCF Wi-Fi Easy Setup, Information technology – Open Connectivity Foundation (OCF)

Specification – Part 7: Wi-Fi Easy Setup specification

Latest version available at:

<https://openconnectivity.org/specs/OCF_Wi-Fi_Easy_Setup_Specification.pdf>

OCF Cloud Specification, Information technology – Open Connectivity Foundation (OCF)

Specification – Part 8: Cloud Specification

Latest version available at:

<https://openconnectivity.org/specs/OCF_Cloud_Specification.pdf>

OCF Cloud Security Specification - Open Connectivity Foundation (OCF) Specification – Cloud

Security Specification

Latest version available at:

<https://openconnectivity.org/specs/OCF_Cloud_Security_Specification.pdf>

OCF Onboarding Tool Specification - Open Connectivity Foundation (OCF) Specification –

Onboarding Tool Specification

Latest version available at:

<https://openconnectivity.org/specs/OCF_Onboarding_Tool_Specification.pdf>

OCF Cloud API for Cloud Services Specification - Open Connectivity Foundation (OCF) Cloud API

for Cloud Services Specification

Latest version available at:

https://openconnectivity.org/specs/OCF\_Cloud\_API\_For\_Cloud\_Services\_Specification.pdf

JSON SCHEMA, draft version 4, [http://json-schema.org/latest/json-schema-core.html.](http://json-schema.org/latest/json-schema-core.html)

IETF RFC 2315, PKCS #7: Cryptographic Message Syntax Version 1.5, March 1998,

<https://tools.ietf.org/html/rfc2315>

IETF RFC 2898, PKCS #5: Password-Based Cryptography Specification Version 2.0, September

2000, <https://tools.ietf.org/html/rfc2898>

IETF RFC 2986, PKCS #10: Certification Request Syntax Specification Version 1.7, November

2000, <https://tools.ietf.org/html/rfc2986>

IETF RFC 4122, A Universally Unique IDentifier (UUID) URN Namespace, July 2005,

https://tools.ietf.org/html/rfc4122

IETF RFC 4279, Pre-Shared Key Ciphersuites for Transport Layer Security (TLS), December

2005, <https://tools.ietf.org/html/rfc4279>

IETF RFC 4492, Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security

(TLS), May 2006, <https://tools.ietf.org/html/rfc4492>

IETF RFC 5246, The Transport Layer Security (TLS) Protocol Version 1.2, August 2008,

<https://tools.ietf.org/html/rfc5246>

IETF RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation

List (CRL) Profile, May 2008, <https://tools.ietf.org/html/rfc5280>

IETF RFC 5489, ECDHE\_PSK Cipher Suites for Transport Layer Security (TLS), March 2009,

<https://tools.ietf.org/html/rfc5489>

IETF RFC 5545, Internet Calendaring and Scheduling Core Object Specification (iCalendar),

September 2009, <https://tools.ietf.org/html/rfc5545>

IETF RFC 5755, An Internet Attribute Certificate Profile for Authorization, January 2010,

<https://tools.ietf.org/html/rfc5755>

IETF RFC 6347, Datagram Transport Layer Security Version 1.2, January 2012,

<https://tools.ietf.org/html/rfc6347>

IETF RFC 6655, AES-CCM Cipher Suites for Transport Layer Security (TLS), July 2012,

<https://tools.ietf.org/html/rfc6655>

IETF RFC 7228, Terminology for Constrained-Node Networks, May 2014,

<https://tools.ietf.org/html/rfc7228>

IETF RFC 7250, Using Raw Public Keys in Transport Layer Security (TLS) and Datagram

Transport Layer Security (DTLS), June 2014, <https://tools.ietf.org/html/rfc7250>

IETF RFC 7251, AES-CCM Elliptic Curve Cryptography (ECC) Cipher Suites for TLS, June 2014,

<https://tools.ietf.org/html/rfc7251>

IETF RFC 7252, The Constrained Application Protocol (CoAP), June 2014,

https://tools.ietf.org/html/rfc7252

IETF RFC 8152, CBOR Object Signing and Encryption (COSE), July 2017,

https://tools.ietf.org/html/rfc8152

IETF RFC 8520, Manufacturer Usage Description Specification, Mar 2019,

<https://tools.ietf.org/html/rfc8520>

IETF RFC 8613, Object Security for Constrained RESTful Environments (OSCORE), July 2019,

https://tools.ietf.org/html/rfc8613

oneM2M Release 3 Specifications, <http://www.onem2m.org/technical/published-drafts>

OpenAPI specification, aka Swagger RESTful API Documentation Specification, Version 2.0

<https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

# 3 Terms, definitions, and abbreviated terms

## 3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in [ISO/IEC 30118-1,](#_bookmark3) [ISO/IEC](#_bookmark4)

[30118-3](#_bookmark4) and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following

addresses:

– ISO Online browsing platform: available at <https://www.iso.org/obp>

– IEC Electropedia: available at <http://www.electropedia.org/>

526 3.1.1

527 Access Management Service (AMS)

528 service that dynamically constructs ACL Resources in response to a Device Resource request

529 Note 1 to entry: An AMS can evaluate access policies remotely and supply the result to a Server which allows or denies

530 a pending access request. An AMS is authorised to provision ACL Resources.

531 3.1.2

532 Credential Management Service (CMS)

533 Device that is authorized to provision credential Resources

534 3.1.3

535 Device Class

536 [IETF RFC 7228](#_bookmark22) defined device class

537 3.1.4

538 Device Ownership Transfer Service (DOTS)

539 logical entity that establishes device ownership

540 3.1.5

541 End-Entity

542 any certificate holder which is not a Root or Intermediate Certificate Authority

543 Note 1 to entry: Typically, a device certificate.

544 3.1.6

545 Intermediary

546 Device that implements both Client and Server roles and may perform protocol translation, virtual

547 device to physical device mapping or Resource translation

548 3.1.7

549 OCF Cipher Suite

550 set of algorithms and parameters that define the cryptographic functionality of a Device. The OCF

551 Cipher Suite includes the definition of the public key group operations, signatures, and specific

552 hashing and encoding used to support the public key.

553 3.1.8

554 OCF Rooted Certificate Chain

555 collection of X.509 v3 certificates in which each certificate chains to a trust anchor certificate which

556 has been issued by a certificate authority under the direction, authority, and approval of the Open

557 Connectivity Foundation Board of Directors as a trusted root for the OCF ecosystem.

558 3.1.9

559 Onboarding Tool (OBT)

560 tool that implements DOTS([3.1.4](#_bookmark34)), AMS([3.1.1](#_bookmark32)), and CMS([3.1.2](#_bookmark33)) functionality

561 3.1.10

562 Out of Band Communication Channel

563 any mechanism for delivery of a secret from one party to another, not specified by OCF

564 3.1.11

Owner Credential (OC)

credential, provisioned to a Device, for the purposes of mutual authentication of the Device and

OBT([3.1.9](#_bookmark35)) during subsequent interactions, identified by having a Subject UUID matching the

Resource Owner Id of the Device Ownership Transfer Resource hosted by a Device that has the

credential

570 3.1.12

571 Role (Network context)

572 stereotyped behavior of a Device; one of [Client, Server or Intermediary]

573 3.1.13

574 Role Identifier

575 Property of an OCF credentials Resource or element in a role certificate that identifies a privileged

576 role that a Server Device associates with a Client Device for the purposes of making authorization

577 decisions when the Client Device requests access to Device Resources.

578 3.1.14

579 Secure Resource Manager (SRM)

580 module in the OCF Core that implements security functionality that includes management of

581 security Resources such as ACLs, credentials and Device owner transfer state.

582 3.1.15

583 Security Virtual Resource (SVR)

584 Resource supporting security features.

585 Note 1 to entry: For a list of all the SVRs please see clause [13.](#_bookmark207)

586 3.1.16

587 Trust Anchor

588 well-defined, shared authority, within a trust hierarchy, by which two cryptographic entities (e.g. a

589 Device and an OBT([3.1.9](#_bookmark35))) can assume trust

590 3.1.17

Device Configuration Resource (DCR)

Resource that is any of the following:

a) a Discovery Core Resource, or

b) a Security Virtual Resource, or

c) a Wi-Fi Easy Setup Resource ("oic.r.easysetup", "oic.r.wificonf", "oic.r.devconf"), or

d) a CoAP Cloud Configuration Resource ("oic.r.coapcloudconf"), or

e) a Software Update Resource ("oic.r.softwareupdate"), or

f) a Maintenance Resource ("oic.wk.mnt").

599 3.1.18

600 Non-Configuration Resource (NCR)

601 Resource that is not a Device Configuration Resource ([3.1.17](#_bookmark37))

602 3.1.19

603 OCF Security Domain

604 set of onboarded OCF Devices that are provisioned with credentialing information for confidential

605 communication with one another

606 3.1.20

607 Owned (or "in Owned State")

608 having the "owned" Property of the "/oic/sec/doxm" Resource equal to "TRUE"

|  |  |
| --- | --- |
| 609 | 3.1.21 |
| 610 | Unowned (or "in Unowned State") |
| 611 | having the "owned" Property of the "/oic/sec/doxm" Resource equal to "FALSE" |
| 612 | 3.1.22 |
| 613 | OCF Onboarding |
| 614 | initial establishment of ownership over a Device, and initial provisioning of the Device for normal |
| 615 | operation |
| 616 | 3.1.23 |
| 617 | Auditable Event |
| 618 | system activity that may be indicative of a violation of security policy |
| 619 | 3.1.24 |
| 620 | Auditable Event Entry |
| 621 | record of the details of an Auditable Event |
| 622 | 3.1.25 |
| 623 | End User |
| 624 | person using the [particular] product |
| 625 | 3.1.26 |
| 626 | End-to-End Secure |
| 627 | securely encapsulate information so that OCF Proxies ([3.1.28](#_bookmark40)) on the end-to-end delivery path do |
| 628 | not need to be trusted with the confidentiality, integrity and freshness of that information |
| 629 | 3.1.27 |
| 630 | End-to-End Security of Unicast Messages |
| 631 | interoperable mechanism which End-to-End Secures the exchange of unicast OCF CRUDN |
| 632 | messages |
| 633 | 3.1.28 |
| 634 | OCF Proxy |
| 635 | functionality which can interpret the OCF compliant URIs of request messages intended for |
| 636 | resources on another OCF Server and can route those request messages accordingly |
| 637 | 3.1.29 |
| 638 | Origin Client |
| 639 | Client which originally generated a request, as opposed to the Client functionality of a Proxy which |
| 640 | is forwarding a request from another Device |
| 641 | 3.1.30 |
| 642 | OSCORE Master Secret |
| 643 | "Master Secret" as defined in clause 3.1 of [IETF RFC 8613](#_bookmark27) |
| 644 | 3.1.31 |
| 645 | OSCORE Recipient ID |
| 646 | "Recipient ID" as defined in clause 3.1 of [IETF RFC 8613](#_bookmark27) |
| 647 | 3.1.32 |
| 648 | OSCORE Security Context |
| 649 | "Security Context" as defined in clause 3.1 of [IETF RFC 8613](#_bookmark27) |
| 650 | 3.1.33 |
| 651 | OSCORE Sender ID |
| 652 | "Sender ID" as defined in clause 3.1 of [IETF RFC 8613](#_bookmark27) |

653 3.1.34

654 OSCORE Sender Sequence Number

655 "Sender Sequence Number" as defined in clause 3.1 of [IETF RFC 8613](#_bookmark27)

656 3.1.35

657 Target Server

658 Server to which a request is addressed, as opposed to the Server functionality of a OCF Proxy

659 ([3.1.28](#_bookmark40)) which receives a request to be forwarded to another Device

660 3.1.36

661 Simple Secure Multicast

662 delivery of UPDATE request messages from a Client to a group of Servers using network-layer

663 multicast, where the messages are protected with a simple security mechanism

664 3.1.37

665 Simple Secure Multicast Client Context

666 OSCORE Security Context ([3.1.32](#_bookmark41)) parameters provisioned to the Client of a Simple Secure 667 Multicast Group ([3.1.38](#_bookmark43)) to enable End-to-End Security of Simple Secure Multicast Requests 668 ([3.1.39](#_bookmark44)) sent to Servers of that Simple Secure Multicast Group ([3.1.38](#_bookmark43))

669 3.1.38

670 Simple Secure Multicast Group

671 group of Servers and one (1) associated Client provisioned with credentials to enable Simple

672 Secure Multicast ([3.1.36](#_bookmark42)) from the Client to the set of Servers

673 3.1.39

674 Simple Secure Multicast Request

675 OSCORE-protected UPDATE request message delivered from a Client to a group of Servers using

676 Simple Secure Multicast ([3.1.36](#_bookmark42))

677 3.1.40

678 Simple Secure Multicast Server Context

679 OSCORE Security Context parameters provisioned to Servers of a Simple Secure Multicast Group 680 ([3.1.38](#_bookmark43)) to enable End-to-End Security of Simple Secure Multicast Requests ([3.1.39](#_bookmark44)) sent by the 681 Client of that Simple Secure Multicast Group ([3.1.38](#_bookmark43))

682 3.1.41

683 Device Onboarding Connection (DOC)

684 special DTLS connection established for the purposes of onboarding the Device securely when a

685 Device is in RFOTM

686 NOTE: The Owner Transfer Method selected will determine the specifics of the DOC used.

687 3.1.42

688 Ready For Normal Operation State

689 state of a Device in which NCRs ([3.1.18](#_bookmark38)) can be accessed

690 3.1.43

691 Ready For Owner Transfer Mechanism State

692 state of a Device in which a Device can be Onboarded

693 3.1.44

694 Ready For Provisioning State

695 state of a Device in which SVRs ([3.1.15](#_bookmark36)) can be configured

696 3.1.45

697 Reset State

698 state of a Device in which the configurable Properties of Device's resources are reset to the

699 manufacturer default and the Device becomes Unowned ([3.1.21](#_bookmark39))

700 3.1.46

Soft Reset State

state of a Device in which SVRs ([3.1.15](#_bookmark36)) can be configured, with slightly more Properties available

than in RFPRO

## 3.2 Symbols and abbreviated terms

AC Access Control

ACE Access Control Entry

ACL Access Control List

AEAD Authenticated Encryption with Authenticated Data

NOTE: Defined in [IETF RFC 8152](#_bookmark25)

AEE Auditable Event Entry

AES Advanced Encryption Standard

AMS Access Management Service

CMS Credential Management Service

COSE CBOR Object Signing and Encryption

NOTE: Defined in [IETF RFC 8152](#_bookmark25)

CRUDN CREATE, RETREIVE, UPDATE, DELETE, NOTIFY

CSR Certificate Signing Request

DOC Device Onboarding Connection

ECC Elliptic Curve Cryptography

ECDSA Elliptic Curve Digital Signature Algorithm

EKU Extended Key Usage

DOTS Device Ownership Transfer Service

ID Identity/Identifier

JSON JavaScript Object Notation.

NVRAM Non-Volatile Random-Access Memory

OC Owner Credential

OCSP Online Certificate Status Protocol

OBT Onboarding Tool

OID Object Identifier

OSCORE

Object Security for Constrained RESTful Environments

NOTE: Defined in

[IETF RFC 8613](#_bookmark27)

OTM - Owner Transfer Method

PE Policy Engine

PIN Personal Identification Number

PPSK PIN-authenticated pre-shared key

PRF Pseudo Random Function

PSI Persistent Storage Interface

PSK Pre Shared Key

RBAC Role Based Access Control

RM Resource Manager

RNG Random Number Generator

RESET Reset State

RFNOP Ready For Normal Operation State

RFOTM Ready For Owner Transfer Mechanism State

RFPRO Ready For Provisioning State

SBAC Subject Based Access Control

SEE Secure Execution Environment

SRESET Soft Reset State

SRM Secure Resource Manager

SSM Simple Secure Multicast

SVR Security Virtual Resource

URI Uniform Resource Identifier

VOD Virtual OCF Device

# 4 Document Conventions and Organization

4.1 Conventions

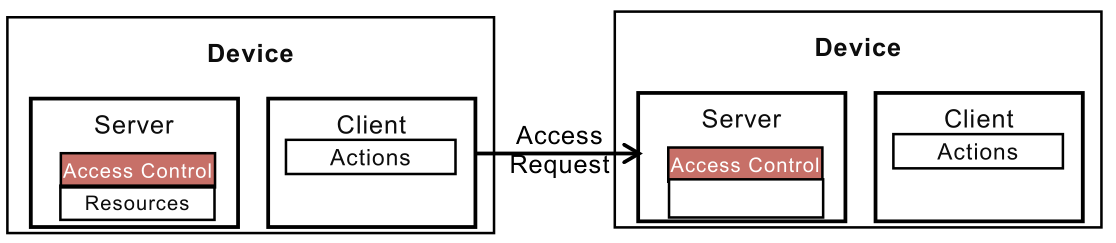
This document defines Resources, protocols and conventions used to implement security for OCF

core framework and applications.

For the purposes of this document, the terms and definitions given in [ISO/IEC 30118-1](#_bookmark3) apply.

In this document, to be consistent with the IETF usages for RESTful operations, the RESTful 760 operation words CRUDN, CREATE, RETRIVE, UPDATE, DELETE, and NOTIFY will have all letters 761 capitalized. Any lowercase uses of these words have the normal technical English meaning.

762 [Figure 1](#_bookmark48) depicts interaction between OCF Devices.



764 Figure 1 – OCF Interaction

765 Devices may implement a Client role that performs Actions on Servers. Actions access Resources 766 managed by Servers. The OCF stack enforces access policies on Resources. End-to-end Device 767 interaction can be protected using session protection protocol (e.g. DTLS) or with data encryption 768 methods.

4.2 Notation

In this document, features are described as required, recommended, allowed or DEPRECATED as

follows:

Required (or shall or mandatory).

These basic features shall be implemented to comply with OCF Core Architecture. The phrases 774 "shall not", and "PROHIBITED" indicate behaviour that is prohibited, i.e. that if performed means 775 the implementation is not in compliance.

776 Recommended (or should).

777 These features add functionality supported by OCF Core Architecture and should be implemented. 778 Recommended features take advantage of the capabilities OCF Core Architecture, usually without 779 imposing major increase of complexity. Notice that for compliance testing, if a recommended 780 feature is implemented, it shall meet the specified requirements to be in compliance with these 781 guidelines. Some recommended features could become requirements in the future. The phrase 782 "should not" indicates behaviour that is permitted but not recommended.

Allowed (may or allowed).

These features are neither required nor recommended by OCF Core Architecture, but if the feature

is implemented, it shall meet the specified requirements to be in compliance with these guidelines.

Conditionally allowed (CA)

The definition or behaviour depends on a condition. If the specified condition is met, then the

definition or behaviour is allowed, otherwise it is not allowed.

Conditionally required (CR)

The definition or behaviour depends on a condition. If the specified condition is met, then the 791 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default 792 unless specifically defined as not allowed.

793 DEPRECATED

794 Although these features are still described in this document, they should not be implemented except 795 for backward compatibility. The occurrence of a deprecated feature during operation of an 796 implementation compliant with the current document has no effect on the implementation’s

797 operation and does not produce any error conditions. Backward compatibility may require that a 798 feature is implemented and functions as specified but it shall never be used by implementations 799 compliant with this document.

Strings that are to be taken literally are enclosed in "double quotes".

Words that are emphasized are printed in italic.

4.3 Data types

See [ISO/IEC 30118-1.](#_bookmark3)

4.4 Document structure

Informative clauses may be found in the Overview clauses, while normative clauses fall outside of

those clauses.

The Security Specification may use the [OpenAPI specification](#_bookmark29) as the API definition language. The

mapping of the CRUDN actions is specified in [ISO/IEC 30118-1.](#_bookmark3)

# 5.Security Overview

## 5.1 Модель безопасности работы

Целью архитектуры безопасности OCF является защита данных и состояний устройств, представленных ресурсами OCF. С точки зрения OCF, Устройство - это сертифицированный логический объект, который участвует в экосистеме OCF. Во время взаимодействия между Устройствами Устройство, выступающее в качестве Сервера, удерживает и контролирует Ресурсы и предоставляет Устройству, выступающему в качестве Клиента, доступ к этим Ресурсам, с учетом набора механизмов безопасности и в соответствии с политиками, настроенными Владельцем домена безопасности OCF. Платформа, на которой размещено Устройство, может обеспечивать усиление безопасности, чтобы гарантировать надежность различных операций, описанных в этом документе. На одной Платформе могут размещаться несколько Устройств.

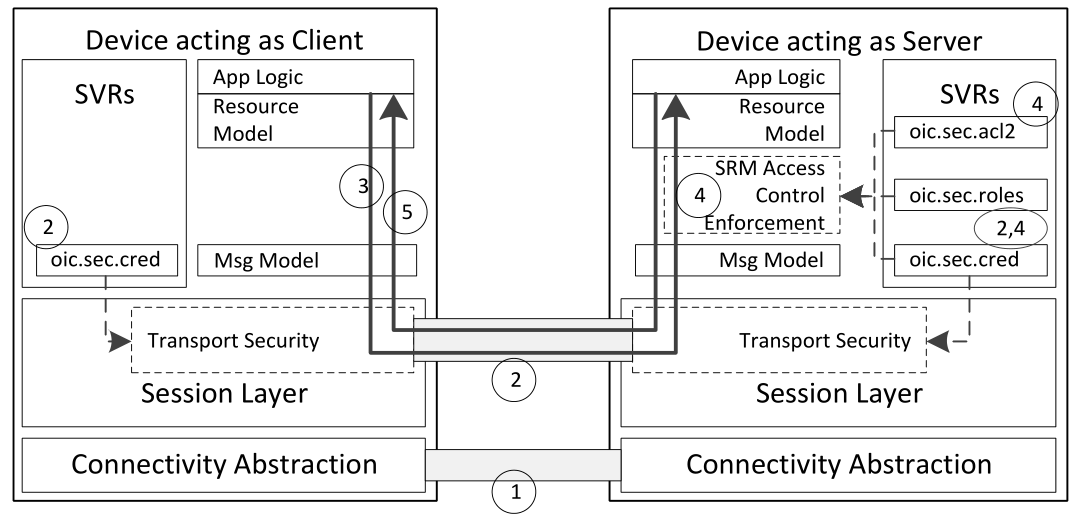
Модель безопасности работы для прямого взаимодействия между устройствами (то есть обмены, которые не поддерживаются объектами, действующими в качестве прокси-серверов OCF между клиентом и сервером) изображена на рисунке 2 и описана в следующих шагах:

Рисунок 2 - Уровни OCF для прямого взаимодействия между устройствами

1. Клиент устанавливает сетевое соединение с Сервером (устройством, содержащим ресурсы).
2. Устройства (сервер и клиент) обмениваются сообщениями либо через защищенный канал с взаимной аутентификацией между двумя Устройствами, либо через незащищенное соединение.
   1. Ресурс «/oic/sec/cred» на каждом Устройстве содержит учетные данные, используемые для взаимной аутентификации, и учетные данные, используемые для авторизации ролей.
   2. Сообщения, полученные по защищенному каналу, связаны с "deviceUUID". В случае удостоверения сертификата «deviceUUID» является частью сертификата, полученного от другого Устройства. В случае учетных данных с симметричным ключом, «deviceUUID» связан с учетными данными в Ресурсе «/oic/sec/cred».
   3. Клиент может представить свой сертификат роли для запроса ассоциации с идентификатором роли ("roleid"). Сервер может связать Клиента с любым количеством идентификаторов ролей.
   4. Запросы, полученные Сервером по незащищенному каналу, рассматриваются как анонимные и не связаны с каким-либо «deviceUUID» или «roleid».
3. Клиент отправляет запрос на Сервер.
4. Сервер получает запрос.
   1. Если запрос получен по незащищенному каналу, Сервер рассматривает запрос как анонимный, и с запросом не связаны «deviceUUID» или «roleid».
   2. Если запрос получен по защищенному каналу, то Сервер по умолчанию связывает запрос с «deviceUUID» Клиента и всеми допустимыми значениями «roleid» Клиента.
   3. Сервер затем просматривает список управления доступом (ACL) и ищет запись управления доступом (ACE), соответствующую следующим критериям:
      1. Запрошенный ресурс соответствует ссылке на ресурс в ACE
      2. Запрошенная операция разрешена "разрешениями" ACE, и
      3. «subjectUUID» содержит либо один из специального набора значений подстановочных знаков, либо, если Устройство не является анонимным, субъект соответствует клиентскому «deviceUUID», связанному с запросом, или действительному «ролевому идентификатору», связанному с запросом. Специальные значения подстановочных знаков разрешают всем Устройствам, обменивающимся данными с аутентифицированными и зашифрованными сеансами или незащищенными сеансами, взаимодействовать в соответствии с ACE.

Если есть соответствующий ACE, то доступ к Ресурсу разрешен; в противном случае доступ запрещен. Доступ обеспечивается диспетчером безопасных ресурсов сервера (SRM).

1. Сервер отправляет ответ клиенту.

OCF также поддерживает обмен сообщениями между клиентом-источником и целевым сервером, осуществляемый одним или несколькими объектами, выступающими в качестве прокси-серверов OCF.

ПРИМЕЧАНИЕ 1. Любое количество прокси-серверов OCF может находиться на пути между исходным клиентом и целевым сервером, хотя на практике ожидается, что это количество будет небольшим.

В некоторых сценариях прокси OCF действует как сервер для входящих сообщений запроса OCF CRUDN: обрабатывает сообщения запроса OCF CRUDN; а затем отправка соответствующих сообщений запроса OCF CRUDN вперед к целевому серверу. Прокси-сервер OCF также может обрабатывать соответствующее входящее ответное сообщение OCF CRUDN и отправлять соответствующие сообщения запроса OCF CRUDN обратно клиенту-источнику.

Этот подход подразумевает, что владелец домена безопасности (содержащий клиент-источник и целевой сервер) готов доверять всем прокси-серверам OCF на пути доставки сообщений конфиденциальность, целостность и актуальность сообщений OCF CRUDN. В качестве альтернативы исходный клиент и целевой сервер могут применять сквозную безопасность одноадресных сообщений, которая позволяет защищать обмен сообщениями OCF CRUDN, так что прокси-серверы OCF не нуждаются в доверии с точки зрения конфиденциальности и целостности сообщений OCF CRUDN.

Модель безопасности работы при использовании прокси OCF без сквозной защиты одноадресных сообщений описана в спецификации облака OCF, спецификации безопасности облака OCF и API C2C.

На рисунках 3 и 4 изображена модель безопасности работы при использовании прокси-серверов OCF и применении сквозной защиты сообщений; см. также следующие шаги. На рисунке 3 показан пример с одним прокси-сервером OCF. На рисунке 4 показан более сложный пример с двумя прокси-серверами OCF, использующими OCF Cloud API для спецификации облачных сервисов; см. примечания 1 и 2.

ПРИМЕЧАНИЕ 2. Если прокси-серверы OCF на рисунке 4 являются облаками OCF, прокси-сервер OCF A - это исходное облако, в котором зарегистрирован исходный клиент, а прокси-сервер OCF B - это целевое облако, в котором зарегистрирован целевой сервер.

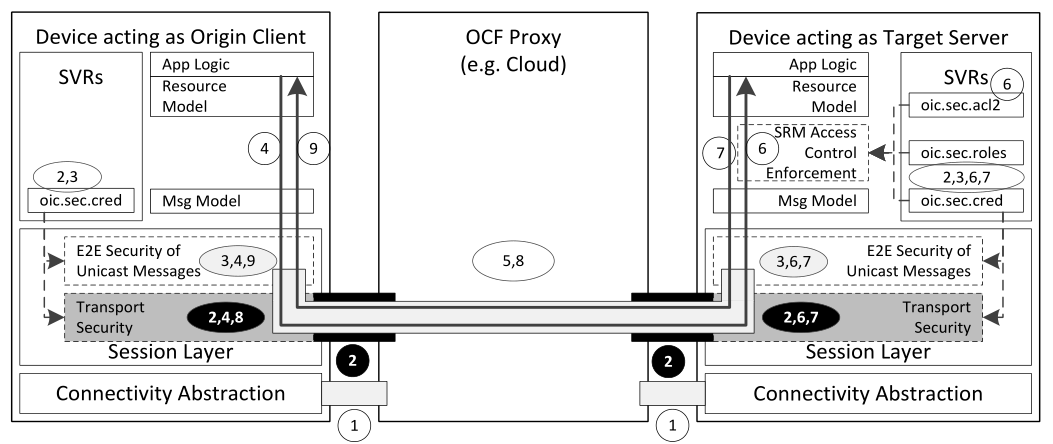


Figure 3 – Уровни OCF для взаимодействия через один прокси-сервер OCF

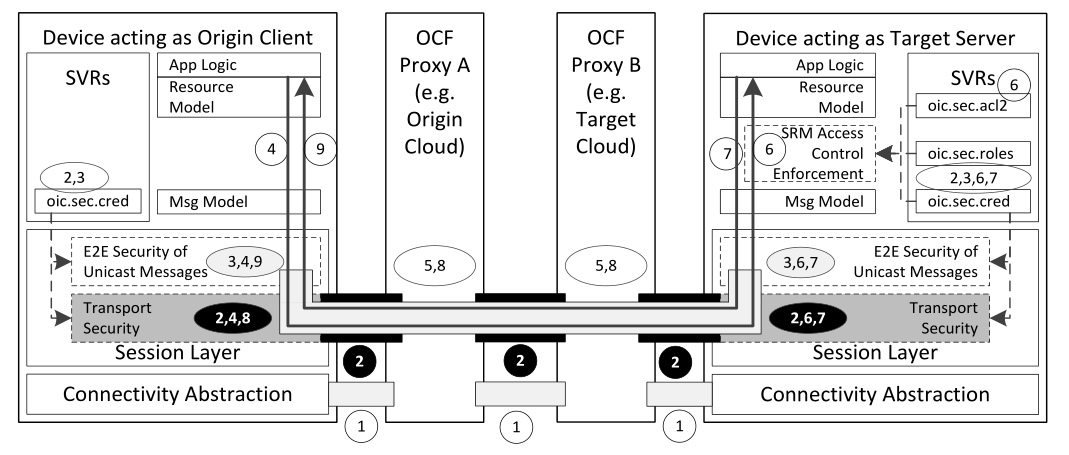
****

Рисунок 4 - Уровни OCF для взаимодействия через два прокси OCF

1) Установлены попарные сетевые соединения.

2) Обмен сообщениями осуществляется по каждому сетевому соединению через попарное безопасное транспортное соединение с взаимной аутентификацией.

3) Исходный клиент и целевой сервер устанавливают сквозной защищенный канал, который взаимно аутентифицируется с использованием учетных данных, содержащихся в ресурсах «/oic/sec/cred» исходного клиента и целевого сервера.

4) Исходный клиент генерирует сообщение запроса OCF CRUDN на целевой сервер. Исходный клиент инкапсулирует сообщение запроса OCF CRUDN в сообщение запроса с сквозной защитой для сквозного защищенного канала (установленного на шаге 3). Информация, идентифицирующая целевой сервер, остается незашифрованной в сообщении запроса End-to-End Secured, поэтому прокси-серверы OCF могут использовать идентифицирующую информацию для правильной маршрутизации сообщения запроса End-to-End Secured. Клиент-источник отправляет сообщение с запросом о сквозной защите на свой прокси-сервер OCF через необязательно защищенное транспортное соединение, установленное с этим прокси-сервером OCF. См. Примечание 3.

5) Каждый прокси-сервер OCF на пути извлекает идентифицирующую информацию о целевом сервере из сообщения запроса и, в соответствии с политиками прокси-сервера OCF, управляющими сообщениями запроса с сквозной защитой, пересылает сквозные защищенные сообщения. сообщение запроса к целевому серверу по необязательно защищенному транспортному соединению. См. Примечания 3, 4 и 5.

6) Целевой сервер проверяет и дешифрует сообщение запроса с сквозной защитой как сообщение канала с сквозной защитой (установленного на шаге 3) для извлечения инкапсулированного сообщения запроса OCF CRUDN от исходного клиента. Сообщение запроса OCF CRUDN рассматривается как полученное через аутентифицированное зашифрованное («auth-crypt») соединение и ассоциированное с «deviceUUID». «DeviceUUID» связан с учетными данными в ресурсе «/oic/sec/cred», используемом для установления сквозного защищенного канала на шаге 3.

7) Целевой сервер определяет, разрешен ли доступ к ресурсу, как описано в шаге 4c модели безопасности для прямого взаимодействия между устройствами, показанной на рисунке 2.

8) Целевой сервер генерирует ответное сообщение OCF CRUDN и инкапсулирует ответное сообщение OCF CRUDN в ответное сообщение с сквозной защитой для сквозного защищенного канала (устанавливается на шаге 3). Target Secure отправляет сообщение ответа End-to-End Secured своему прокси-серверу OCF через необязательно защищенное транспортное соединение, по которому был получен соответствующий запрос. См. Примечание 3.

9) Каждый прокси-сервер OCF на пути пересылает сообщение ответа End-to-End Secured клиенту-источнику через необязательно защищенное транспортное соединение, по которому было получено соответствующее сообщение запроса. См. Примечание 3.

10) Исходный клиент проверяет и расшифровывает ответное сообщение с сквозной защитой как сообщение сквозного защищенного канала (установленного на шаге 3) для извлечения инкапсулированного ответного сообщения OCF CRUDN с целевого сервера.

ПРИМЕЧАНИЕ 3: Во время передачи сообщение OCF CRUDN может быть защищено двумя независимыми уровнями безопасности: уровнем сквозной безопасности одноадресных сообщений (с использованием OSCORE) и независимым уровнем безопасности транспорта (с использованием DTLS). или TLS).

ПРИМЕЧАНИЕ 4. В этом документе не рассматриваются подробности того, как прокси OCF определяет, разрешает ли его политика пересылку сообщения запроса на идентифицированный целевой сервер. Если прокси-сервер OCF разрешает пересылку сообщения запроса на целевой сервер, то предполагается, что прокси-сервер OCF также разрешает пересылку соответствующего ответного сообщения (сообщений) по транспортному соединению, по которому было получено соответствующее сообщение запроса.

ПРИМЕЧАНИЕ 5. В этом документе не рассматривается, как OCF Proxy A определяет, что OCF Proxy B является правильным OCF Proxy для пересылки сообщения запроса. Спецификация OCF Cloud API для облачных сервисов предоставляет подробные сведения для случая, когда прокси-сервер OCF A и прокси-сервер OCF B являются облаками OCF.

Как показано на рисунке 5, Simple Secure Multicast (SSM) позволяет Клиенту безопасно передавать запрос UPDATE группе серверов с помощью одного неподтвержденного запроса UPDATE, доставленного через многоадресную рассылку сетевого уровня.

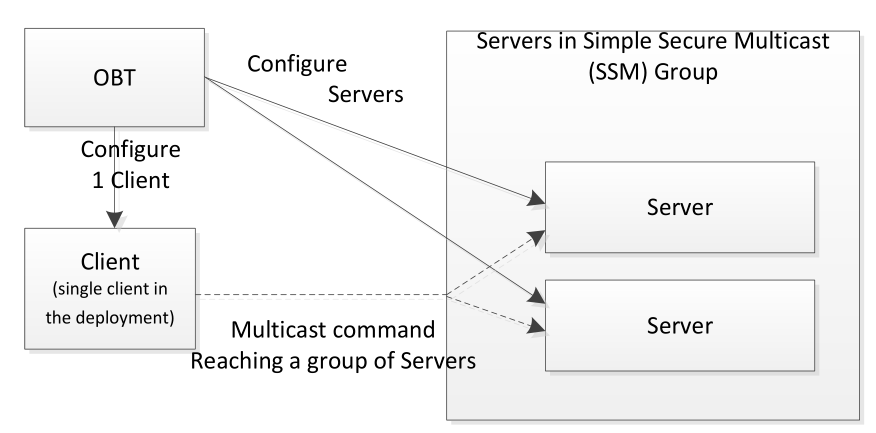


Рисунок 5 - Один запрос достигает группы серверов

Модель безопасности для SSM описана на рисунке 6 и сопровождающих его этапах.

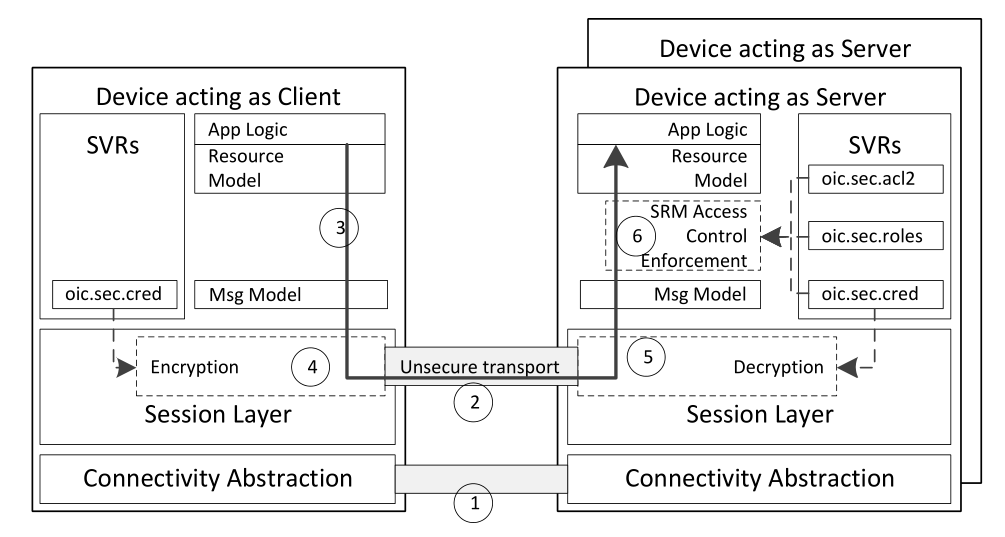


Рисунок 6 - Уровни OCF для простой безопасной многоадресной рассылки

1. Клиент и серверы в группе SSM настроены с шифрованием / дешифрованием. Клиент знает заранее сконфигурированный многоадресный адрес для использования и знает, как создать фактическую полезную нагрузку команды для отправки.
2. Обмен сообщениями осуществляется через незащищенное транспортное соединение. Клиент генерирует сообщение запроса UPDATE для серверов.
3. Клиент инкапсулирует сообщение запроса UPDATE в сообщение запроса End-to-End Secured незащищенного канала. Адрес многоадресной рассылки остается незашифрованным в защищенном сообщении запроса.
4. Клиент отправляет защищенное сообщение запроса UPDATE на URL-адрес многоадресной рассылки Серверов, используя URL-адрес ресурса с включенной многоадресной рассылкой.
5. Серверы расшифровывают сообщение. Сообщение с запросом UPDATE рассматривается как полученное через аутентифицированное зашифрованное («auth-crypt») соединение и связано с «deviceUUID» (который может быть UUID устройства Клиента).
6. Сервер определяет, разрешен ли доступ к Ресурсу, как описано в шаге 4c модели безопасности для прямого взаимодействия между устройствами, показанной на рисунке 2.

Защита ресурсов включает защиту данных как в состоянии покоя, так и во время передачи. Помимо механизмов контроля доступа, Спецификация безопасности OCF не включает спецификацию безопасного хранения ресурсов. Безопасное хранение может быть достигнуто за счет использования аппаратной защиты или шифрования данных в состоянии покоя. Точная реализация безопасного хранилища подчиняется ряду ужесточающих требований, указанных в пункте 14, и может быть предметом руководящих принципов сертификации.

Защита данных при передаче полностью определена как нормативная часть этого документа. Этот документ поддерживает защиту данных при передаче данных на транспортном уровне за счет использования таких механизмов, как DTLS и сквозная защита данных при передаче через OSCORE.

ПРИМЕЧАНИЕ 6. DTLS будет обеспечивать защиту пакета за пакетом, а не защиту сообщения OCF CRUDN в целом. Например, если требуется целостность всего сообщения OCF CRUDN в целом, перед передачей пакета на транспортный уровень следует применить отдельную сквозную безопасность (например, с использованием OSCORE).

На рисунке 7 показаны точки обеспечения безопасности OCF.

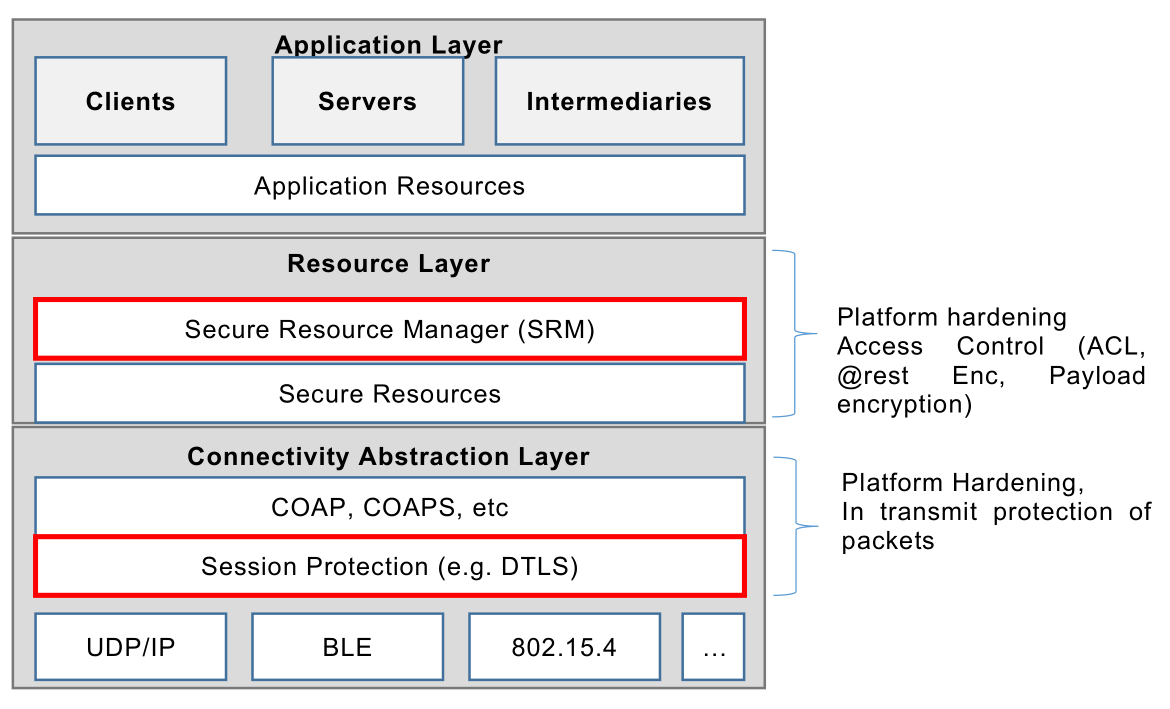


Рисунок 7 - Точки обеспечения безопасности OCF

## 5.2 Контроль доступа

### 5.2.1 Общие сведения о контроле доступа

Инфраструктура OCF предполагает, что Ресурсы размещаются на Сервере и доступны Клиентам с учетом механизмов контроля доступа и авторизации. Ресурсы на Сервере защищены посредством реализации контроля доступа, аутентификации и защиты конфиденциальности.

В этом разделе представлен обзор управления доступом с помощью списков управления доступом.

Однако управление доступом в OCF не зависит от уровней абстракции транспорта и связности.

Реализация контроля доступа основывается на априорном определении набора политик доступа к Ресурсу. Политики хранятся локально в ресурсе ACL, предоставляемом службой управления доступом (AMS) в форме записей управления доступом (ACE). Отсутствие такого связанного ACE приводит к недоступности Ресурса. Могут применяться несколько типов механизмов контроля доступа:

- Управление доступом на основе субъекта (SBAC), где ACE сопоставляет идентичность Клиента с субъектом, включенным в политику, определенную для Ресурса. Для подтверждения личности Клиента требуется процесс аутентификации.

- Управление доступом на основе ролей (RBAC), где ACE сопоставляет идентификатор роли, включенный в политику для ресурса, с идентификатором роли, связанным с клиентом.

- Управление доступом на основе подстановочных знаков, когда ACE соответствует типу соединения, используемому для доступа к Ресурсу (то есть любому соединению с взаимной аутентификацией).

ACE применяется только в том случае, если ACE соответствует как субъекту (т. Е. Клиенту), так и запрошенному ресурсу.

Существует несколько способов сопоставления объекта: (1) UUID устройства, (2) идентификатор роли или (3) подстановочный знак. Способ, которым Клиент подключается к Серверу, может иметь значение для принятия решений по управлению доступом. Сопоставление подстановочных знаков при подключении с проверкой подлинности и без проверки подлинности, а также с использованием зашифрованного и незашифрованного соединения позволяет широко применять политику доступа к предметным классам.

Пример политики сопоставления с подстановочными знаками:

"aclist2": [

{

"subject": {"conntype": "anon-clear"},

"resources": [

{ "wc": "\*" }

],

"permission": 31

},

{

"subject": {"conntype": "auth-crypt" },

"resources": [

{ "wc": "\*" }

],

"permission": 31

}

]

Подробности формата ACL определены в разделе 12. ACL состоит из одного или нескольких ACE.

Некоторые ресурсы, такие как коллекции, генерируют запросы к связанным ресурсам при использовании соответствующих интерфейсов. В таких случаях необходимо учитывать дополнительные аспекты управления доступом.

Дополнительные соображения по управлению доступом для Коллекций при использовании пакетного интерфейса OCF можно найти в пункте 12.2.7.3. Ресурс ACL требует такой же защиты, как и другие важные

Ресурсы, когда речь идет как о хранении, так и о работе с SRM.

### 5.2.2 Архитектура ACL

Сервер проверяет ресурсы, запрошенные клиентом, перед обработкой запроса. Ресурс управления доступом ищется, чтобы найти одну или несколько записей ACE, которые соответствуют Клиенту и запрошенным ресурсам. Если совпадение найдено, применяются ограничения разрешения и периода.

Если найдено более одного совпадения, то каждая запись ACE оценивается на совпадение независимо.

Сервер использует контекст соединения, чтобы определить, аутентифицирован ли субъект или нет, и была ли применена конфиденциальность данных. Если пользователь прошел аутентификацию, то соответствие субъектов может происходить с повышенной детализацией на основе идентификации роли или устройства.

Каждый ACE содержит набор разрешений, который будет применяться к данному Клиенту. Разрешения состоят из комбинации действий CREATE, RETREIVE, UPDATE, DELETE и NOTIFY (CRUDN).

Клиенты аутентифицируются как Устройство и могут работать с одной или несколькими ролями. При утверждении роли устройства могут получать повышенные права доступа. Например, роль «oic.role.owner» может предоставлять дополнительные ресурсы и интерфейсы OCF, которые обычно недоступны.

Ресурсы ACL размещаются на серверах локально. Локальные ACL обеспечивают большую автономию в обработке контроля доступа.

Следующие варианты использования описывают работу контроля доступа:

Сценарий использования 1. Как показано на рисунке 8, на серверном устройстве размещено 4 ресурса (R1, R2, R3 и R4). Клиентское устройство D1 запрашивает доступ к ресурсу R1, размещенному на серверном устройстве 5. ACL [0] соответствует ресурсу R1 и включает D1 в качестве авторизованного субъекта. Таким образом, Устройство D1 получает доступ к Ресурсу R1, потому что локальный ACL «/ oic / sec / acl2 / 0» соответствует запросу..

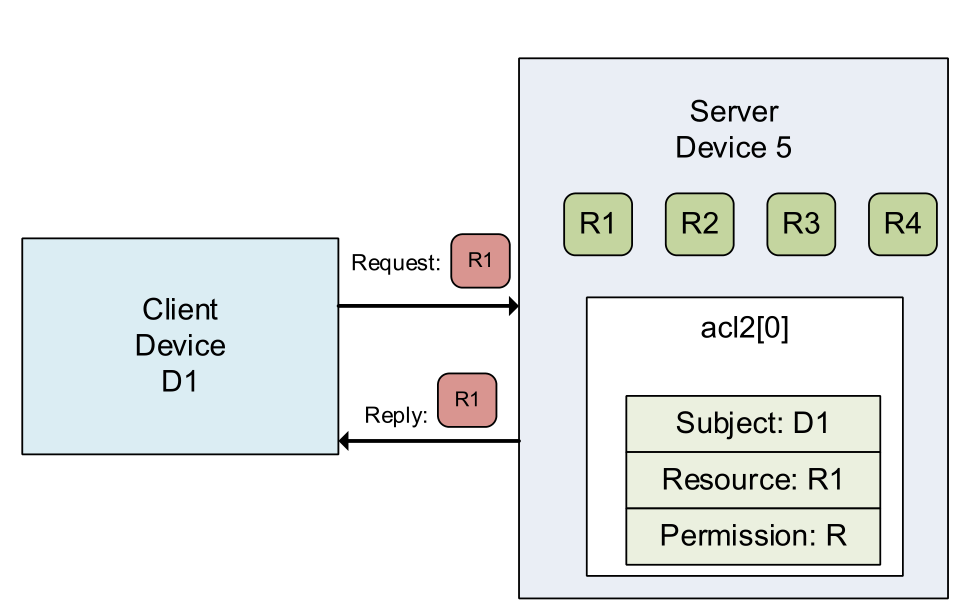


Рисунок 8 - Пример использования 1, показывающий простое применение ACL

## 5.3 Обзор адаптации

### 5.3.1 Общие сведения о подключении

Прежде чем устройство станет работоспособным в среде OCF и сможет взаимодействовать с другими устройствами, оно должно быть соответствующим образом подключено. Первым шагом в подключении Устройства является настройка владения, при которой законный пользователь, который владеет / покупает Устройство, использует инструмент подключения (OBT), а использование OBT использует один из методов передачи владельца (OTM) для установления права собственности. Как только право собственности установлено, OBT подготавливает Устройство, после чего Устройство становится работоспособным и может взаимодействовать с другими Устройствами в среде OCF.

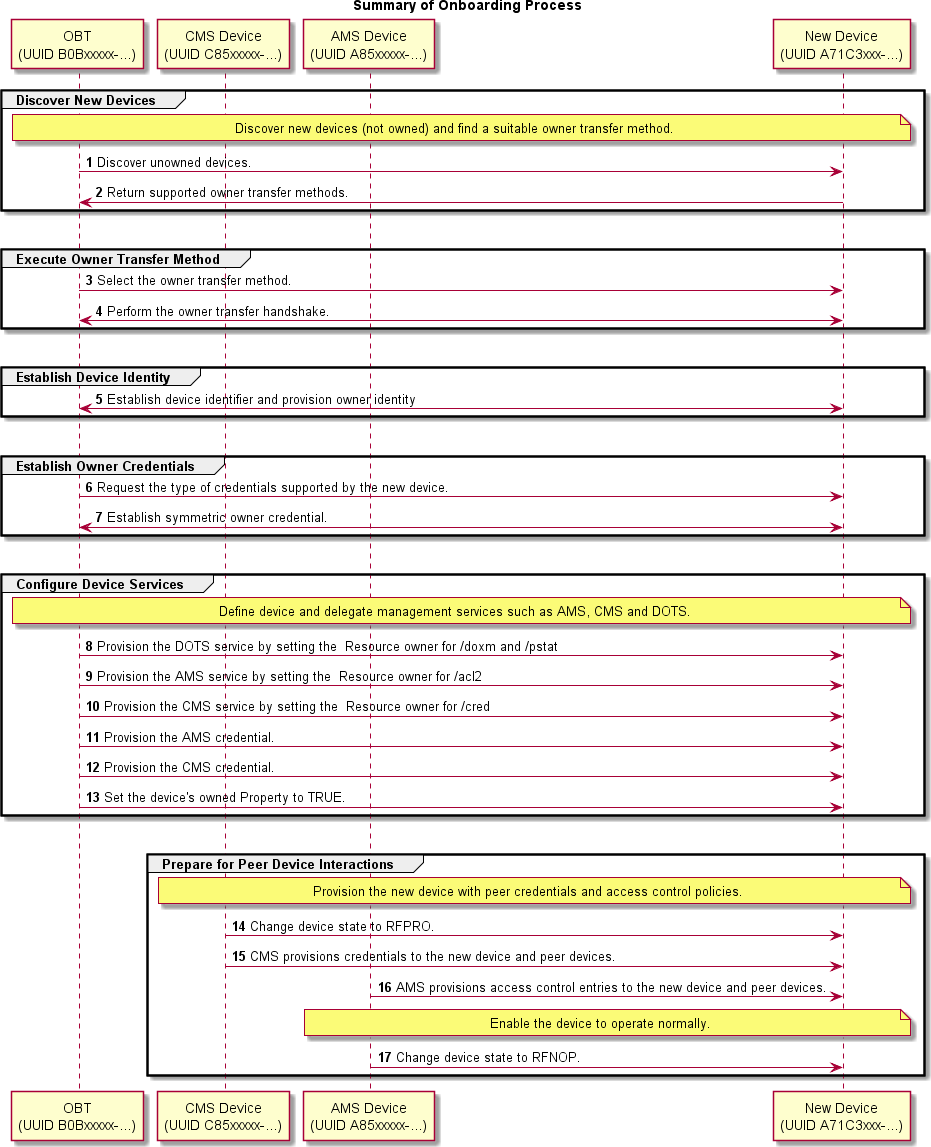
На рисунке 9 показан обзор адаптации.

Рисунок 9 - Обзор адаптации

В этом пункте объясняется процесс подключения и обеспечения безопасности, но предоставление не связанных с безопасностью аспектов оставляет на усмотрение других документов OCF. В контексте безопасности все Устройства должны иметь минимальную конфигурацию безопасности, которая позволяет Устройству безопасно взаимодействовать / связываться с другими Устройствами в среде OCF. Эта минимальная конфигурация безопасности определяется как RFNOP для встроенного устройства и указана в 8.

## 5.3.2 Этапы адаптации

Блок-схема на рисунке 10 показывает типичные шаги, которые выполняются во время адаптации. Хотя подключение может включать в себя множество шагов, не связанных с безопасностью, основное внимание на диаграмме уделяется конфигурации, связанной с безопасностью, чтобы новое устройство могло функционировать в среде OCF. Подключение обычно начинается с того, что Устройство становится Собственным устройством, за которым следует настройка Устройства для среды, в которой оно будет работать. Это будет включать в себя информацию о настройке, например, кто может получить доступ к Устройству и какие действия могут быть выполнены, а также какие разрешения у Устройства для взаимодействия с другими Устройствами.

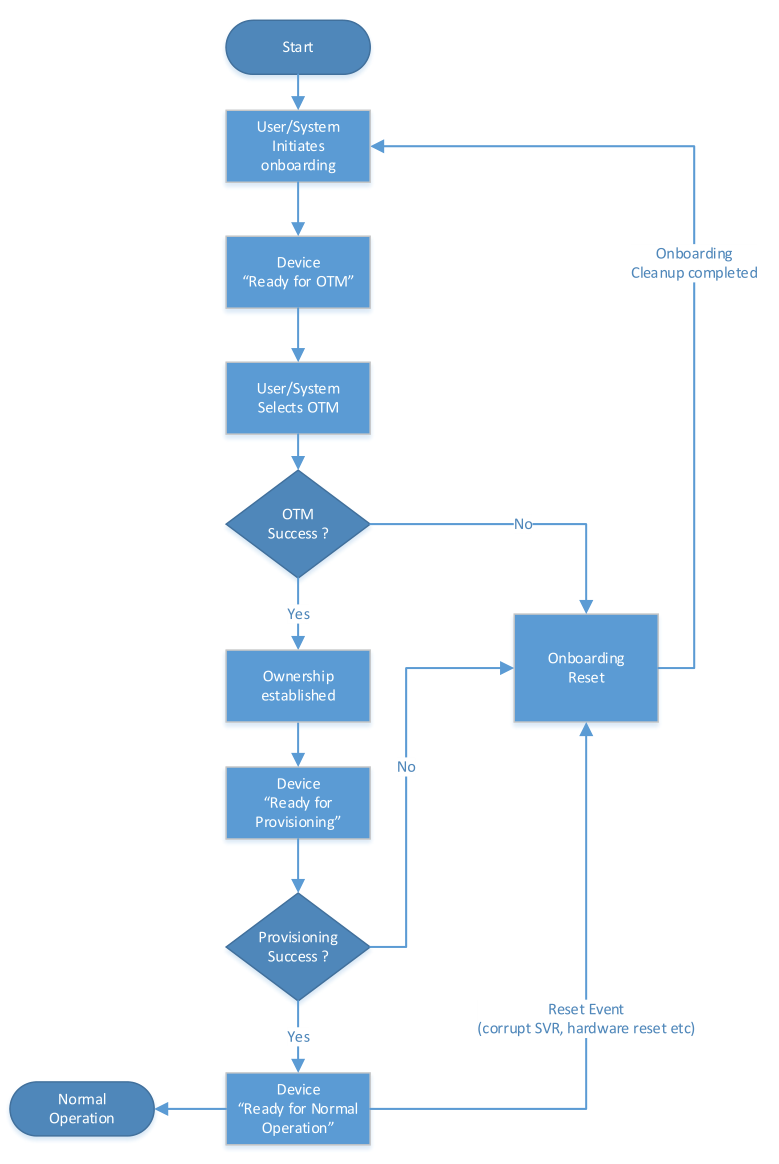


Рисунок 10 - Процесс адаптации OCF

### 5.3.3. Установление владельца устройства

Цель установления владения устройством - позволить владельцу домена безопасности OCF заявить о себе в качестве владельца и менеджера устройства и ввести Устройство в домен безопасности OCF. Это достигается за счет использования DOTS, который включает создание контекста владения между новым Устройством и DOTS и обеспечивает оперативный контроль и управление Устройством. DOTS размещается на OBT.

DOTS использует один из OTM, указанных в 7.3, для безопасного установления владения устройством.

OTM устанавливает нового владельца (оператора DOTS), который уполномочен управлять устройством.

Передача права собственности выполняет следующее:

- DOTS предоставляет учетные данные владельца (OC) для свойства "creds" в ресурсе "/ oic / sec / cred" устройства. Этот OC позволяет Устройству и DOTS взаимно аутентифицироваться во время последующих взаимодействий. OC связывает UUID устройства DOTS с "rowneruuid"

Свойство ресурса "/oic/sec/doxm", устанавливающее его как владельца ресурса.

- Владелец устройства устанавливает доверие к устройству через OTM.

- Предоставление соответствующих учетных данных для устройства, которое должно быть членом домена безопасности OCF.

### 5.3.4 Подготовка к нормальной работе

После того, как Устройство получит необходимую информацию для инициализации, следующим шагом будет предоставление дополнительной конфигурации безопасности, которая позволит Устройству начать работу. Это может включать в себя настройку различных параметров, а также может включать несколько шагов. Также в это время выполняется подготовка списков ACL для различных ресурсов, размещенных на сервере на устройстве. Этап подготовки не ограничивается только этим этапом. Инициализация устройства может происходить на нескольких этапах жизненного цикла устройства. Однако на этом этапе, скорее всего, произойдет конкретное обеспечение состояния ресурсов и свойств, связанное с безопасностью, в конце которого каждое Устройство достигает RFNOP. RFNOP согласован и четко определен независимо от конкретного используемого OTM или независимо от того, что предоставляется. Однако отдельные механизмы OTM и этапы подготовки могут указывать дополнительную конфигурацию состояний ресурсов и свойств. Минимальная обязательная конфигурация, необходимая для того, чтобы Устройство находилось в RFNOP, указана в 8.

### 5.3.5 Система управления соответствием OCF

OCF Compliance Management System (OCMS) - это сервис, поддерживаемый OCF, который предоставляет статус сертификации и информацию для устройств OCF.

OCMS предоставляет список сертифицированных продуктов (CPL) в формате JSON, размещенный по адресу URI: https://www.openconnectivity.org/certification/ocms-cpl.json.

OBT должен иметь корневой сертификат, необходимый для включения https-соединения с URI https://www.openconnectivity.org/certification/ocms-cpl.json.

OBT должен периодически обновлять свою копию CPL через URI https://www.openconnectivity.org/certification/ocms-cpl.json в соответствии с требованиями политики владельца домена безопасности OCF.

## 5.4 Подготовка

### 5.4.1 Общие положения

Обеспечение безопасности OCF включает в себя процессы во время и после передачи права собственности, такие как конфигурация учетных данных для взаимодействия со службами обеспечения, настройка любых ресурсов, связанных с безопасностью, и учетных данных для взаимодействия с любыми службами или устройствами, с которыми настроенное Устройство должно связаться позже.

Устройство должно взаимодействовать с CMS и AMS, чтобы обеспечить:

* Учетные данные безопасности через CMS, который в настоящее время предполагается встроенным в тот же OBT, что и DOTS.
* Политики контроля доступа и ACL через AMS, который в настоящее время предполагается встроенным в тот же OBT, что и DOTS.

Чтобы иметь возможность поддерживать использование отдельных служб управления устройствами, некоторые защищенные виртуальные ресурсы устройства (SVR) имеют связанного владельца ресурса, указанного в свойстве rowneruuid ресурса.

Свойство "rowneruuid" ресурсов "/oic/sec/doxm" и "/ oic / sec / pstat" идентифицирует DOTS..

Свойство "rowneruuid" ресурса "/ oic / sec / cred" идентифицирует CMS.

Свойство «rowneruuid» ресурса «/ oic / sec / acl2» идентифицирует AMS.

DOTS предоставляет учетные данные, которые обеспечивают безопасные соединения между службами OCF и новым устройством. DOTS инициирует управляемую клиентом инициализацию, сигнализируя службе OCF.

### 5.4.2 Обеспечение контроля доступа

Подготовка ACL выполняется через безопасное соединение между AMS и его устройствами. AMS подготавливает ACL, обновляя ресурс ACL устройства.

### 5.4.3 Предоставление учетных данных

CMS безопасно предоставляет учетные данные для взаимодействия между устройствами, используя учетные данные CMS, предоставленные DOTS во время процедуры подключения. Также ожидается, что CMS будет проактивно отслеживать учетные данные, установленные на Устройстве, и обновлять их при необходимости (например, ближе к дате истечения срока действия).

### 5.4.4 Предоставление ролей

Серверы, получающие запросы на ресурсы, которые они размещают, должны проверить идентификаторы ролей, заявленные Клиентом, запрашивающим Ресурс, и сравнить эти идентификаторы ролей с ограничениями, описанными в списках ACL Сервера Таким образом, Клиенту может потребоваться с одним или несколькими учетными данными для ролей. После предоставления клиент может утверждать роль, которую он использует, как описано в 10.4.2, если у него есть учетные данные роли сертификата.

Каждое Устройство содержит информацию о утверждаемых ролях как Свойство в Ресурсе учетных данных.

Каждое устройство хранит информацию о заявленных ролях как свойства в ресурсе ролей.

Все заявленные роли используются при применении ACL. Когда сервер имеет несколько ролей, заявленных для Клиента, доступ к Ресурсу предоставляется, если он будет предоставлен под любой из ролей.

## 5.5 Менеджер защищенных ресурсов (SRM)

SRM играет ключевую роль в обеспечении общей безопасности. Короче говоря, SRM выполняет как управление SVR, так и контроль доступа для запросов на доступ и управление ресурсами. SRM состоит из 3 основных функциональных элементов:

- Менеджер ресурсов (RM): отвечает за 1) загрузку SVR из постоянного хранилища (с использованием PSI) по мере необходимости. 2) Обеспечение механизма политик (PE) ресурсами по запросу. 3) Ответы на запросы SVR. В то время как SVR находятся в памяти SRM, SVR имеют формат, совместимый с форматом хранилища данных для конкретного устройства. Однако RM будет использовать формат JSON для упорядочивания структур данных SVR перед их передачей в PSI для хранения или перемещением за пределы устройства.

- Механизм политик (PE), который принимает запросы на доступ к SVR и на основе политик управления доступом отвечает на запросы либо «ACCESS\_GRANTED», либо «ACCESS\_DENIED». Чтобы принять решение о доступе, PE обращается к соответствующему ACL и ищет лучший элемент управления доступом (ACE), который может обслуживать запрос с учетом субъекта (устройства или роли), который был аутентифицирован DTLS.

- Интерфейс постоянного хранилища (PSI): PSI предоставляет набор API-интерфейсов для RM для управления файлами в его собственной памяти и хранилище. Дизайн SRM является модульным, поэтому он может быть реализован в безопасной среде выполнения Платформы; если доступно.

На рисунке 11 изображена архитектура SRM OCF.

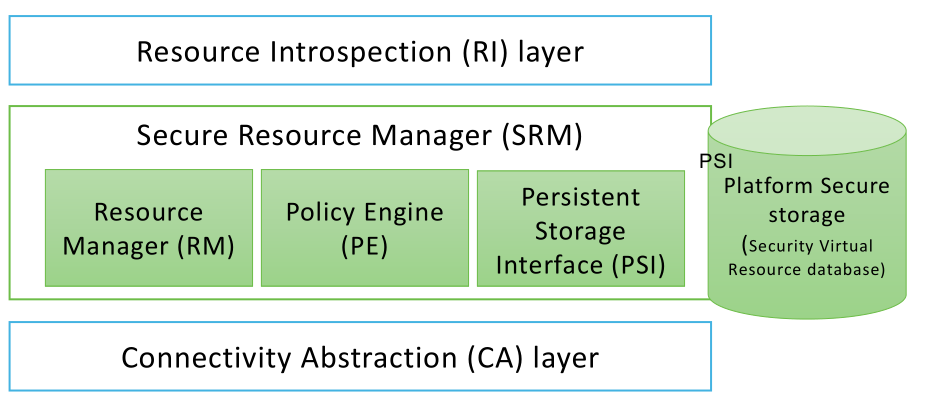


Рисунок 11 - Архитектура SRM OCF

## 5.6 Обзор учетных данных

Устройства могут использовать учетные данные для подтверждения личности и роли (ролей) сторон в связи между клиентом и сервером. Полномочия могут быть симметричными или асимметричными. Каждое Устройство хранит секретные и общедоступные части своих собственных учетных данных, где это применимо, а также учетные данные для других Устройств, которые были предоставлены DOTS или CMS. Эти учетные данные затем могут использоваться при установлении сеансов защищенной связи (например, с использованием DTLS, TLS или OSCORE). Ролевые сертификаты могут использоваться после установления сеанса с аутентификацией для утверждения одной или нескольких ролей для Устройства.

В этом документе доступны следующие типы учетных данных:

* Парные симметричные ключи
* Сертификаты
* Сырые асимметричные ключи

Устройства могут не поддерживать все эти типы учетных данных. Набор поддерживаемых типов учетных данных для любого Устройства содержится в его Свойстве «sct» ресурса «/oic/sec/doxm».

## 5.7 Регистрация событий

### 5.7.1 Общие сведения о регистрации событий

Платформа OCF может генерировать различные виды контролируемых событий. Эти контролируемые события можно использовать для анализа журнала или для понимания состояния системы в реальном времени. Обычно сохраняется несколько контролируемых событий для отслеживания проблем, возникших в системе. Емкость хранилища IoT-устройств обычно очень ограничена, поэтому часто используется определенный тип структуры данных, такой как кольцевой буфер.

Устройство OCF регистрирует записи контролируемых событий (AEE) для всех контролируемых событий, которые удовлетворяют свойствам «categoryfilter» и «priorityfilter» ресурса «/ oic / sec / ael». AEE хранятся в локальном хранилище (см. Рисунок 1). Из-за ограниченного размера локального хранилища ожидается, что владелец домена безопасности OCF настроит параметры фильтрации.

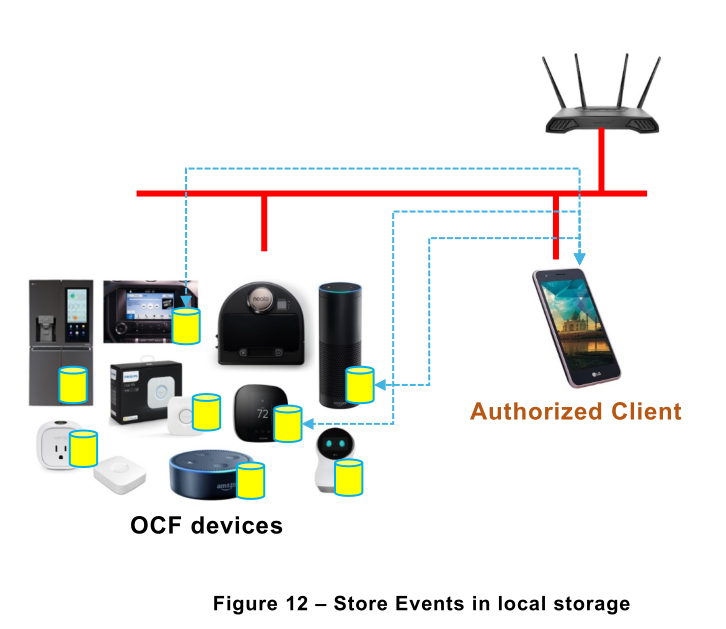


Рисунок 12 - Сохранение событий в локальном хранилище

## 5.8 Сквозная безопасность одноадресных сообщений

Модель безопасности для сквозной защиты одноадресных сообщений описана на рисунках 3 и 4 пункта 5.1 и сопутствующих этапах.

OCF использует протокол безопасности объектов для сред с ограничениями RESTful (OSCORE) IETF RFC 8613 для сквозной защиты одноадресных сообщений. Исходный клиент преобразует сообщение запроса OCF CRUDN в кодировке CoAP в сообщение запроса OSCORE, которое может быть переадресовано на целевой сервер прокси-серверами OCF; Затем целевой сервер обрабатывает сообщение запроса OSCORE, чтобы извлечь сообщение запроса OCF CRUDN. Аналогичным образом целевой сервер затем преобразует закодированное с помощью CoAP сообщение ответа OCF CRUDN в сообщение ответа OSCORE, которое может быть переадресовано клиенту-источнику прокси-серверами OCF; Происхождение

Затем клиент обрабатывает ответное сообщение OSCORE, чтобы извлечь ответное сообщение OCF CRUDN. OSCORE сохраняет конфиденциальность, целостность и актуальность сообщений OCF CRUDN при передаче между исходным клиентом и целевым сервером.

Спецификация OSCORE поддерживает транспортировку сообщений OSCORE с использованием протокола CoAP, уже используемого в спецификациях OCF. Полезная нагрузка сообщения OSCORE - это объект подписи и шифрования объекта CBOR (COSE) (см. IETF RFC 8152), в котором все элементы сообщения OCF CRUDN с кодировкой CoAP, кроме тех частей, которые необходимы для доставки сообщения получателю Устройство зашифрованы и целостность защищена. OSCORE также включает защиту от воспроизведения.

## 5.9 Обзор Simple Secure Multicast

Модель безопасности для SSM описана на рисунке 6 пункта 5.1 и сопутствующих этапах. Для обеспечения безопасности сообщений SSM протокол OSCORE IETF RFC 8613 использует протокол OSCORE. Клиент преобразует закодированное в CoAP сообщение запроса UPDATE в сообщение запроса OSCORE, которое может быть направлено на серверы группы SSM с использованием многоадресной рассылки сетевого уровня; Затем Сервер обрабатывает сообщение запроса OSCORE, чтобы извлечь сообщение запроса UPDATE.

Примечание. OSCORE также используется, хотя и немного иначе, для сквозной защиты одноадресных сообщений.

Предполагаемое использование функции SSM - только для обновления ресурсов с помощью одного неподтвержденного многоадресного запроса. Другие операции CRUDN (например, RETRIEVE, подтверждаемое ОБНОВЛЕНИЕ и т. Д.) Не поддерживаются, поскольку протокол SSM не предназначен для отправки индивидуальных ответов на запрос. Следовательно, при отправке такой операции с помощью SSM отдельные Серверы будут молча игнорировать сообщение запроса и не отправлять ответ.

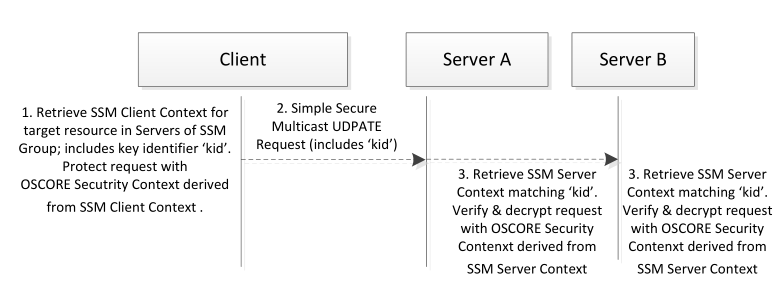
Спецификация OSCORE поддерживает транспортировку сообщений OSCORE с использованием протокола CoAP, уже используемого в спецификациях OCF. Полезная нагрузка сообщения OSCORE - это объект CBOR Object Signing and Encryption (COSE) (см. IETF RFC 8152), в котором все элементы сообщения запроса UPDATE с кодировкой CoAP, кроме тех частей, которые необходимы для доставки сообщения получателю Устройство зашифрованы и целостность защищена. OSCORE также включает защиту от воспроизведения.

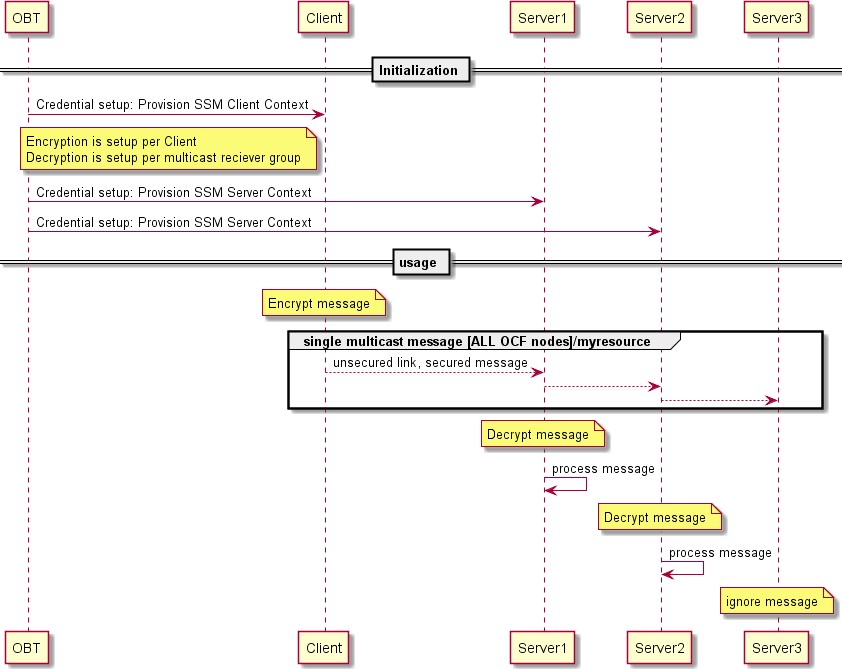
Настройка контекста безопасности OSCORE для группы SSM представляет собой отношение 1-N:

- Клиентский контекст SSM группы SSM предоставляется только один раз в клиенте группы SSM, и

- копии контекста сервера SSM группы SSM предоставляются одному или нескольким серверам в группе SSM.

На рисунке 13 изображена взаимосвязь контекста клиента SSM и контекста сервера SSM.



Рисунок 13 - Диаграмма взаимосвязи для сообщений Simple Secure Multicast

На рисунке 14 показана полная настройка и использование.

Рисунок 14 - Настройка и использование Secure Simple Multicast

Первому сообщению после подключения сервер неявно доверяет как действительное сообщение. Это связано с тем, что окно воспроизведения еще не настроено Сервером. Сервер сохраняет полученную информацию, так что защита от воспроизведения включается после получения первого сообщения.

# 6. Безопасность процесса обнаружения

## 6.1 Преамбула

Основная функция механизма обнаружения заключается в предоставлении универсальных идентификаторов ресурсов (URI, называемых ссылками) для ресурсов, размещенных на сервере, дополненных атрибутами этих ресурсов и возможными дальнейшими связями. (в соответствии с разделом 10 ISO / IEC 30118-1)

## 6.2 Соображения безопасности при обнаружении

При определении процесса обнаружения необходимо позаботиться о том, чтобы обнаруживающему объекту был предоставлен только минимальный набор ресурсов без нарушения безопасности конфиденциальной информации или требований конфиденциальности рассматриваемого приложения. Сюда входят как данные, включенные в Ресурсы, так и соответствующие метаданные.

Для достижения расширяемости и масштабируемости в этом документе не указывается возможность обнаружения каждого отдельного ресурса. Вместо этого Сервер, содержащий Ресурс, будет полагаться на ACL для каждого Ресурса, чтобы определить, имеет ли запрашивающая сторона (Клиент) право просматривать / обрабатывать какие-либо ресурсы.

Ресурс «/ oic / sec / acl2» содержит записи ACL, управляющие доступом к ресурсам, размещенным на сервере. (См. 13.5)

Помимо конфиденциальности и возможности обнаружения ресурсов с точки зрения ACL, сам процесс обнаружения должен быть защищен. Этот документ устанавливает следующие требования для процесса обнаружения:

1) Обеспечение защиты целостности обнаруженных ресурсов.

2) Обеспечение защиты конфиденциальности для обнаруженных ресурсов, которые считаются конфиденциальными.

Обнаружение ресурсов осуществляется с помощью операции RETRIEVE (одноадресной или многоадресной)

на известном Ресурсе "/ oic / res".

Запрос на обнаружение отправляется по незащищенному каналу (многоадресная или одноадресная без DTLS), Сервер не может определить личность запрашивающего. В таких случаях Сервер, который хочет аутентифицировать Клиента перед ответом, может перечислить URI безопасного обнаружения (например, coaps: // IP: PORT / oic / res) в ответе незащищенного ресурса «/ oic / res». Это означает, что URI безопасного обнаружения по умолчанию доступен для обнаружения любым Клиентом. Затем от клиента потребуется отправить отдельный одноадресный запрос с использованием DTLS на URI безопасного обнаружения.

Например, клиент с UUID устройства «d1» (UUID: «0685B960-736F-46F7-BEC0-9E6CBD61ADC1») делает запрос RETRIEVE для ресурса «/ door», размещенного на сервере с UUID устройства «d3», где d3 имеет ACL2

{

"aclist2": [

{

"subject": {"uuid": "0685B960-736F-46F7-BEC0-9E6CBD61ADC1"},

"resources": [{"href":"/door"}],

"permission": 2, // RETRIEVE

"aceid": 1

},

{

"subject": {"authority": "owner", "role": "owner"}

"resources": [{"href":"/door"}],

"permission": 2, // RETRIEVE

"aceid": 2

},

{

"subject": {"uuid": "0685B960-736F-46F7-BEC0-9E6CBD61ADC1"},

"resources": [{"href":"/door/lock"}],

"permission": 4, // UPDATE

"aceid": 3

}

],

"rowneruuid": "0685B960-736F-46F7-BEC0-9E6CBD61ADC1"

}

UUID устройства "d3", где d3 имеет ACL2:

ACL указывает, что Клиент «d1» имеет разрешения RETRIEVE для ресурса. Следовательно, когда устройство «d1» обнаруживает ресурс «/ door» на сервере «d3», ответ будет включать все URI из ресурса «/ door». Клиент «d2» без идентификатора роли «владелец» получит ответ об ошибке, который не содержит URI.

Результаты обнаружения, доставленные в d1 относительно ресурса d3 "/ door" из защищенного интерфейса:

[

{

"href": "/door",

“rel”: “self”,

"rt": ["oic.wk.col"],

"if": ["oic.if.ll", “oic.if.b”, "oic.if.baseline"],

"eps":[{"ep": "coaps://[2001:db8:a::b1d4]:55555"}]

},

{

"href": "/door/lock",

"rt": ["oic.r.lock.status "],

"if": ["oic.if.a", “oic.if.baseline”],

"eps":[{"ep": "coaps://[2001:db8:a::b1d4]:55555"}]

}

]

# 7 Обеспечение безопасности

## 7.1 Идентификация устройства

### 7.1.1 Общая идентификация устройства

Устройство должно идентифицироваться значением UUID устройства, которое устанавливается как часть подключения устройства и содержится в свойстве «deviceuuid» ресурса «/oic/sec/doxm». UUID устройства должны быть уникальными в рамках соответствующего домена безопасности OCF, и ожидается, что они будут генерироваться случайным образом и обеспечиваться OBT. Ожидается, что DOTS проверит, что выбранный новый UUID устройства не конфликтует с UUID устройства, ранее введенными в домен безопасности OCF.

Устройства поддерживают связь своих UUID устройств и собственных криптографических учетных данных через Ресурс «/ oic / sec / cred». Идентификационные данные связаны криптографически в случае учетных данных сертификата или связаны через внутренние сопоставления в Ресурсе «/ oic / sec / cred» в противном случае. Ресурс «/ oic / sec / cred» поддерживает список собственных и других учетных данных Устройства. Несколько учетных данных могут быть связаны с одним и тем же UUID устройства. Ожидается, что устройство будет представлять учетные данные, связанные с его собственным UUID устройства, только для целей аутентификации однорангового узла. Устройства рассматривают ресурс «/ oic / sec / cred» как авторитетный при проверке учетных данных для аутентификации однорангового Устройства.

В случае аутентифицированного соединения UUID устройства рассматривается как идентификатор клиента для целей проверки контроля доступа для целевого ресурса. UUID устройства клиента сопоставляется с UUID субъекта в предварительно подготовленных записях ресурса сервера "/ oic / sec / acl2". Сервер определяет UUID клиентского устройства на основе учетных данных, используемых для установления сеанса.

Платформа OCF, на которой может размещаться несколько Устройств, идентифицируется идентификатором платформы. Идентификатор платформы является глобально уникальным и вставляется в устройство с защитой целостности (например, в защищенном хранилище или подписан и проверен).

Платформа OCF может иметь безопасную среду выполнения, используемую для защиты уникальных идентификаторов и секретов. Если на платформе размещено несколько Устройств, необходим некоторый механизм, чтобы предоставить каждому Устройству соответствующий и отдельный контекст безопасности.

### 7.1.2 Идентификация устройства для устройств с UAID [устарело]

Этот пункт намеренно оставлен пустым.

## 7.2 Владение устройством

Это информативный пункт. Устройства - это логические объекты, которые являются конечными точками безопасности, имеющими удостоверение, которое можно проверить с помощью криптографических учетных данных. При первой инициализации устройство не является владельцем. Установление права собственности на устройство - это процесс, с помощью которого устройство заявляет свою идентичность в DOTS, а DOTS предоставляет идентичность владельца. Этот обмен приводит к тому, что устройство меняет свое состояние владения, тем самым не позволяя другой DOTS устанавливать административный контроль над устройством.

Процесс передачи права собственности начинается с того, что OBT обнаруживает новое устройство, которое находится в состоянии Unowned, путем проверки свойства «Owned» ресурса «/oic/sec/doxm» нового устройства. По окончании перехода права собственности выполняется следующее:

1) DOTS устанавливает безопасный сеанс с новым устройством.

2) Необязательно утверждает любое из следующего:

а) Близость (с использованием ПИН-кода) OBT к Платформе.

б) Сертификат производителя, подтверждающий поставщика платформы, модель и другие атрибуты платформы.

3) Определяет идентификатор устройства.

4) Определяет владельца устройства.

5) Указывает владельца устройства (например, UUID устройства OBT).

6) Предоставляет устройству учетные данные владельца.

7) Устанавливает состояние «Owned» нового устройства в TRUE.

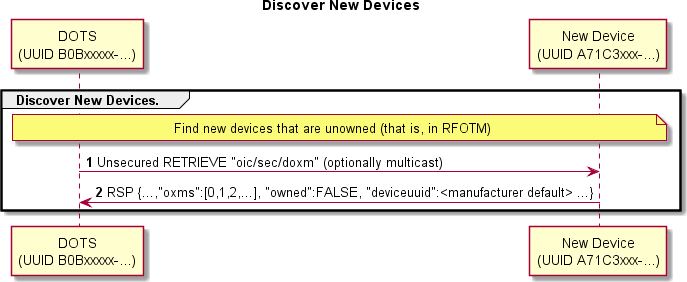
## 7.3 Методы передачи права собственности на устройство

### {7.3.1 Требования к реализации OTM

В этом документе представлены спецификации для нескольких методов передачи права собственности. Внедрение каждого индивидуального метода передачи прав собственности считается необязательным. Однако каждое устройство должно реализовывать по крайней мере один из методов передачи права собственности, не включая методы, специфичные для поставщика.

Все OTM, включенные в этот документ, считаются необязательными. Каждый поставщик должен выбрать и реализовать по крайней мере один из OTM, указанных в этом документе. Тем не менее, OCF предполагает, что будут существовать подходы, ориентированные на конкретного поставщика. Если поставщик желает иметь возможность взаимодействия между OTM конкретного поставщика и OBT других поставщиков, он должен работать напрямую с поставщиками OBT для обеспечения взаимодействия. Тем не менее, стандартизация ОТМ является предпочтительным подходом. В таких случаях в 7.3.7 предоставляется набор руководящих принципов, помогающих поставщикам в разработке OTM для конкретных поставщиков.

Ресурс «/oic/sec/doxm» является расширяемым, чтобы использовать методы передачи владельца (OTM), определенные поставщиком. DOTS определяет, какой OTM лучше всего подходит для установки на новое устройство.

Все OTM должны представлять возможности подключения Устройства с использованием свойства «oxms» ресурса «/oic/sec/doxm». DOTS определяет поддерживаемые типы учетных данных устройства с помощью свойства «sct» поддерживаемых типов учетных данных ресурса «/oic/sec/doxm». Учетные данные предоставления DOTS и CMS в соответствии с поддерживаемыми типами учетных данных.

На рисунке 15 изображена последовательность обнаружения нового устройства.

Figure 15 – Discover New Device Sequence

Table 1 – Discover New Device Details

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | DOTS запрашивает, не принадлежит ли еще новое устройство. |
| 2 | Новое устройство возвращает ресурс «/oic/sec/doxm», содержащий статус владения и поддерживаемые OTM. Он также содержит временный UUID устройства, который может измениться после успешной передачи владельца. Устройство должно предоставить временный идентификатор, чтобы облегчить обнаружение в качестве гостевого устройства.  Обратитесь к спецификации OCF Onboarding Tool для ознакомления с соображениями безопасности при выборе OTM. |

Устройство должно поддерживать выборочное использование незащищенной многоадресной рассылки для получения запросов RETRIEVE к Ресурсу устройства «/oic/sec/doxm», как показано на рисунке 15. В пункте 10.4 стандарта ISO / IEC 30118-1 представлены общие сведения об использовании многоадресной рассылки CoAP. запросы в OCF. Получение многоадресной рассылки ресурса "/oic/sec/doxm" поддерживает фильтрацию с использованием параметра запроса "own". Когда многоадресный запрос RETRIEVE пропускает параметр запроса «принадлежит» или включает параметр запроса «принадлежащий», установленный на «ложь», тогда Устройство должно отвечать только в том случае, если Устройство находится в RFOTM и нет открытого подключения для подключения устройства. В противном случае запрос будет проигнорирован Устройством, независимо от конфигурации ACE.

OTM устройств конкретного производителя должны соответствовать Спецификации ресурсов "/oic/sec/doxm" для OC, которая является результатом OTM устройства конкретного поставщика. OTM конкретного поставщика должен включать положения для установления доверия к новому Устройству со стороны DOTS и, при необходимости, установления доверия к OBT с помощью нового Устройства.

Новому устройству может потребоваться выполнить некоторые шаги инициализации в начале OTM. Например, если инициируется OTM на основе случайного PIN-кода, новое устройство может сгенерировать случайное значение PIN. DOTS обновляет свойство oxmsel «/oic/sec/doxm» до значения, соответствующего используемому OTM, перед выполнением других шагов OTM. Это обновление уведомляет новое устройство о начале передачи права собственности.

Конечное состояние OTM, зависящего от производителя, должно позволить новому Устройству аутентифицироваться в OBT, а OBT - в новом устройстве.

Дополнительные шаги инициализации могут быть выполнены после успешного переноса владельца с использованием установленного сеанса OTM.

### 7.3.2 Расчет учетных данных SharedKey

Учетные данные SharedKey выводятся с использованием PRF, который принимает значение key\_block, полученное в результате рукопожатия DTLS, используемого для подключения. Новое Устройство должно использовать следующий расчет для обеспечения взаимодействия между продуктами поставщика (DOTS выполняет тот же расчет):

SharedKey = PRF(Secret, Message);

Where:

* PRF shall use TLS 1.2 PRF defined by [IETF RFC 5246](#_bookmark15) clause 5.
* Secret - это key\_block, полученный в результате рукопожатия DTLS.
  + See [IETF RFC 5246](#_bookmark15) clause 6.3
  + Длина key\_block зависит от набора шифров.
    - (e.g. 96 bytes for TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256 40 bytes for TLS\_PSK\_WITH\_AES\_128\_CCM\_8)
* Сообщение представляет собой конкатенацию следующего:
  + Строка DoxmType для текущего метода подключения (например, "oic.sec.doxm.jw")
    - См. Пункт 13.2.2 для конкретных типов DoxmTypes.
  + Owner ID - это UUID, идентифицирующий идентификатор владельца устройства и устройство, которое поддерживает SharedKey.
    - Используйте необработанные байты, как указано в IETF RFC 4122, пункт 4.1.2.
  + UUID устройства - это UUID нового устройства.
    - Используйте необработанные байты, как указано в IETF RFC 4122, пункт 4.1.2.
* Длина SharedKey будет 32 октета.
  + Если в последующих сеансах DTLS используются комплекты шифров 128-битного шифрования, будут использоваться крайние левые 16 октетов. Сеансы TLS с использованием комплектов шифров с 256-битным шифрованием будут использовать все 32 октета.

### 7.3.3 Создание учетных данных сертификата

Учетные данные сертификата будут использоваться Устройствами для безопасной двунаправленной связи. Сертификаты будут выданы CMS или внешним центром сертификации (CA). Этот CA будет использоваться для взаимной проверки подлинности Устройства.

### 7.3.4 Just-Works OTM

#### 7.3.4.1 Just-Works OTM Общие

Just-works OTM создает учетные данные с симметричным ключом, который представляет собой предварительный общий ключ, используемый для установления безопасного соединения, через которое устройство должно быть подготовлено для использования в домене безопасности OCF владельца. Предоставление дополнительных учетных данных и ресурсов - типичный шаг после установления права собственности. Общий ключ называется SharedKey.

DOTS выбирает OTM Just-works, используя свойство «oxmsel» ресурса «/oic/sec/doxm», и устанавливает сеанс DTLS, используя набор шифров, определенный для Just-works OTM.

Последовательность Just Works OTM показана на рисунке 16, а шаги описаны в таблице 2.

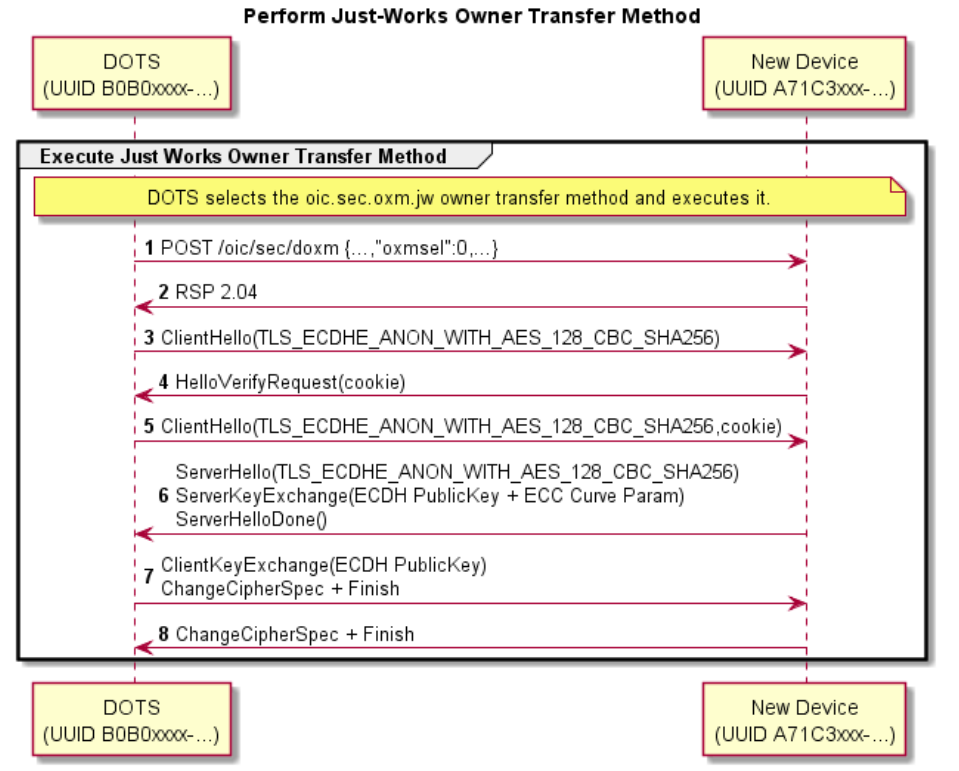


Рисунок 16 - Программа Just Works OTM

Таблица 2 - Подробная информация об A Just Works OTM

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1, 2 | DOTS уведомляет устройство о том, что оно выбрало метод "Just Works".. |
| 3 - 8 | Сессия DTLS устанавливается с использованием анонимного протокола Диффи-Хеллмана..a |
| a Этот метод предполагает, что оператор осведомлен о возможности атаки «злоумышленник посередине» и принял меры предосторожности для выполнения этого метода в сети «чистых помещений». | |

#### 7.3.4.2 Соображения безопасности

Анонимное соглашение о ключах Диффи-Хеллмана подвержено атаке типа "злоумышленник посередине". Использование этого метода предполагает, что и DOTS, и новое устройство выполняют метод «просто работает», предполагая, что подключение происходит в относительно безопасной среде, в отсутствие атакующего устройства.

У этого метода нет надежного способа подтвердить, что заявленный UUID устройства надежно привязан к устройству.

Новое устройство должно использовать временный UUID устройства до перехода на собственное устройство, в то время как оно считается гостевым устройством, чтобы предотвратить отслеживание с учетом конфиденциальности. Устройство утверждает невременной UUID устройства, который может отличаться от временного значения во время защищенного сеанса, в котором происходит обмен передачей владельца. DOTS проверяет, что заявленный UUID устройства не конфликтует с уже используемым UUID устройства. Если он уже используется, существующие учетные данные используются для установления безопасного сеанса.

Не принадлежащее ему Устройство, для которого также установлены учетные данные устройства, может указывать на поврежденное или взломанное устройство.

### 7.3.5 OTM на основе случайного PIN-кода

#### 7.3.5.1 Общие положения OTM на основе случайного PIN-кода

Метод случайного ПИН-кода устанавливает физическую близость между новым устройством и OBT, что позволяет предотвратить атаки типа «злоумышленник в середине». Устройство генерирует случайное число, которое передается в DOTS по внеполосному каналу связи. Определение внеполосного канала связи выходит за рамки определения OTM устройств. DOTS и новое Устройство используют PIN-код при обмене ключами в качестве доказательства того, что Конечный пользователь санкционировал передачу права собственности, имея физический доступ к новому Устройству через канал внеполосной связи.

#### 7.3.5.2 Последовательность смены владельца на основе случайного ПИН-кода

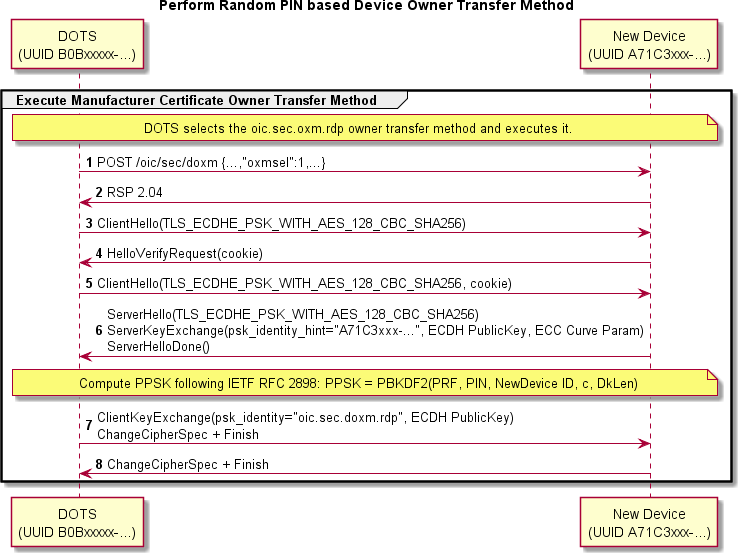
Случайная последовательность OTM на основе PIN-кода показана на рисунке 17, а шаги описаны в таблице 3.

Рисунок 17 - OTM на основе случайного PIN-кода

Таблица 3 - Детали OTM на основе случайных ПИН-кодов

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1, 2 | DOTS уведомляет Устройство о том, что оно выбрало метод «Случайный PIN». |
| 3 - 8 | Сеанс DTLS устанавливается с использованием набора шифров Диффи-Хеллмана на основе PSK. PIN-код предоставляется как параметр PSK. ПИН-код генерируется случайным образом новым устройством, а затем передается через канал внеполосной связи, который устанавливает ближайший контекст между новым устройством и DOTS. Принцип безопасности заключается в том, что атакующее устройство не сможет перехватить PIN-код из-за отсутствия близости. |

К сообщениям подтверждения DTLS для этого OTM применяются следующие требования:

- На шаге 6:

- Сервер должен использовать только набор шифров DTLS, поддерживаемый OTM на основе случайного ПИН-кода (см. Пункт 11.3.2.2),

- Новое Устройство должно установить поле «psk\_identity\_hint» сообщения ServerKeyExchange на конкатенацию

- строка «oic.sec.doxm.rdp»;

- символ двоеточия ':';

- Свойство "deviceuuid" ресурса "/oic/sec/doxm" отправляется в ответах, когда новое Устройство находится в RFOTM и когда подключение для подключения устройства в настоящее время не установлено.

- На шаге 7:

- Если новое Устройство определяет, что поле «psk\_identity» сообщения ClientKeyExchange не совпадает со строкой «oic.sec.doxm.rdp», то новое Устройство должно отклонить подтверждение DTLS.

- новое Устройство будет применять приведенное ниже определение ключа.

ПРИМЕЧАНИЕ Строка «oic.sec.doxm.rdp» является URN, определенным для OTM на основе случайного PIN в таблице 18, и включена, чтобы позволить будущим OTM повторно использовать наборы шифров DTLS без путаницы в отношении того, какой OTM следует применять.

Этот OTM использует псевдослучайную функцию (PBKDF2), определенную IETF RFC 2898, и PIN-код, передаваемый через внеполосный канал связи, для генерации предварительного общего ключа. Предварительный общий ключ с аутентификацией PIN (PPSK) предоставляется в комплекты шифров TLS, которые принимают PSK.

- PPSK = PBKDF2 (PRF, PIN, UUID устройства, c, dkLen)

Функция PBKDF2 имеет следующие параметры:

- PRF - использует TLS 1.2 PRF, определенный IETF RFC 5246.

- ПИН - получен через внеполосный канал связи.

- UUID устройства - свойство «deviceuuid» ресурса «/oic/sec/doxm», отправляемое в ответах, когда новое Устройство находится в RFOTM и когда подключение для подключения устройства в настоящее время не установлено.

Используйте необработанные байты, как указано в пункте 4.1.2 IETF RFC 4122.

- c - Счетчик итераций инициализирован до 1000

- dkLen - Желаемая длина производного PSK в октетах.

#### 7.3.5.3 Соображения безопасности

Безопасность механизма случайного ПИН-кода зависит от энтропии ПИН-кода. Использование ПИН-кода с недостаточной энтропией может позволить атаке «злоумышленник-посередине» восстановить любые долгосрочные учетные данные, предоставленные как часть адаптации. В частности, изучение предоставленных учетных данных с симметричным ключом позволяет злоумышленнику маскироваться под встроенное устройство.

Рекомендуется, чтобы энтропия ПИН была достаточной, чтобы противостоять онлайн-атаке грубой силы, 40 бит или более. Например, 12-значный цифровой PIN-код, или 8-значный буквенно-цифровой (0-9a-z), или 7-значный буквенно-цифровой PIN-код с учетом регистра (0-9a-zA-Z). Атака «человек посередине» - это когда злоумышленник активен в сети и может перехватывать и изменять сообщения между DOTS и устройством. При атаке «злоумышленник посередине» злоумышленник должен восстановить PIN-код из сообщений обмена ключами в «реальном времени», то есть до тайм-аута однорангового узла, и прервать попытку подключения.

Восстановив ПИН-код, он может завершить этап аутентификации обмена ключами. Приведенное здесь руководство требует минимум 40 бит энтропии, однако гарантия, которую это дает, зависит от ресурсов, доступных злоумышленнику. Учитывая параллелизируемый характер атаки методом грубой силы, атака имеет линейное ускорение по мере добавления большего количества ядер / потоков. Более консервативная величина энтропии составила бы 64 бита. Поскольку Random PIN OTM требует использования набора шифров DTLS, который включает обмен ключами ECDHE, безопасность Random PIN OTM всегда по крайней мере эквивалентна безопасности JustWorks OTM.

В Random PIN OTM также есть возможность использовать PBKDF2 для извлечения ключевого материала из PIN. Обоснование состоит в том, чтобы увеличить стоимость атаки методом перебора путем увеличения стоимости каждого предположения в атаке на настраиваемую величину (количество итераций PBKDF2). Теоретически это эффективный способ снизить требования к энтропии PIN. К сожалению, это сокращение трудно определить количественно, поскольку увеличение времени, затрачиваемого честными партнерами в X раз, напрямую не приводит к увеличению времени атакующего в X раз. Эта асимметрия возникает из-за того, что злоумышленник может использовать специализированные реализации и оборудование, недоступные для честных партнеров. По этой причине при принятии решения о том, сколько энтропии использовать для ПИН-кода, разработчикам рекомендуется исходить из того, что PBKDF2 не обеспечивает безопасности, и гарантировать, что ПИН-код имеет достаточную энтропию.

Безопасность OTM устройства произвольного ПИН-кода зависит от предположения, что существует защищенный канал внеполосной связи для передачи случайно сгенерированного ПИН-кода от нового устройства к OBT. Если канал внеполосной связи передает злоумышленнику часть или весь ПИН-код, это снижает энтропию ПИН-кода, и применяются атаки, описанные выше. Канал внеполосной связи следует выбирать таким образом, чтобы он требовал близости между DOTS и новым устройством. Предполагается, что злоумышленник не скомпрометировал внешний канал связи. В качестве примера канала внеполосной связи устройство может отображать PIN-код, который необходимо ввести в программное обеспечение OBT. Другой пример - устройство кодирует PIN-код как 2D-штрих-код и отображает его для камеры на устройстве DOTS для захвата и декодирования.

### 7.3.6 OTM на основе сертификата производителя

#### 7.3.6.1 Общие положения OTM на основе сертификата производителя

OTM на основе сертификата производителя должен использовать сертификат, встроенный в устройство производителем, и может использовать подписанный OBT, который определяет якорь доверия между устройством и DOTS.

Встроенные сертификаты производителя не обязательно должны связываться с якорем доверия корневого CA OCF.

Для некоторых сред, политик или администраторов может потребоваться дополнительная информация о характеристиках устройства. Этот список дополнительных аттестаций, которые OCF могла или не могла тестировать (понимая, что некоторые аттестации не могут быть протестированы или для которых тестирование может быть неосуществимым или экономически невыгодным), можно найти в расширении OCF Security Claims x509.v3, описанном в 9.4.2.2. .6.

При использовании передачи владения на основе сертификатов устройства должны использовать асимметричные ключи с данными сертификатов для аутентификации своей личности с помощью DOTS в процессе ввода нового устройства в действие в домене безопасности OCF. Процесс адаптации состоит из нескольких отдельных этапов:

1) Предварительные условия на борту

a) Элемент учетных данных Ресурса учетных данных Устройства («/ oic / sec / cred»), содержащий сертификат производителя, должен быть идентифицирован свойством «credusage», содержащим строку «oic.sec.cred.mfgcert», чтобы указать, что учетные данные содержит сертификат производителя.

б) Цепочка сертификатов производителя должна содержаться в свойстве "publicdata" идентифицированного элемента учетных данных.

c) Устройство должно содержать уникальную и неизменяемую пару асимметричных ключей ECC.

d) Если устройство требует аутентификации DOTS в рамках передачи права собственности, предполагается, что DOTS была зарегистрирована и получила сертификат для своей уникальной и неизменной пары асимметричных ключей ECC, подписанный заранее заданным якорем доверия.

e) Конечный пользователь настроил приложение DOTS, указав информацию о доступе к сети и информацию об учетной записи (если есть).

2) DOTS аутентифицирует Устройство с помощью ECDSA для проверки подписи. Кроме того, Устройство может аутентифицировать DOTS для проверки подписи DOTS.

3) Если аутентификация не удалась, Устройство должно указать причину сбоя и вернуться в RFOTM. Если аутентификация прошла успешно, Устройство должно установить зашифрованное соединение с DOTS в соответствии с согласованным набором шифров.

#### 7.3.6.2 Профили сертификатов

Подробности см. В 9.4.2.

#### 7.3.6.3 Вопросы безопасности последовательности передачи владельца сертификата

OBT должен аутентифицировать устройство во время посадки. Устройство не аутентифицирует OBT.

Во время установления связи DTLS сервер не должен отправлять запрос сертификата.

#### 7.3.6.4 Последовательность ОТМ на основе сертификата производителя

Последовательность OTM на основе сертификата производителя показана на рисунке 18, а шаги описаны в таблице 4.

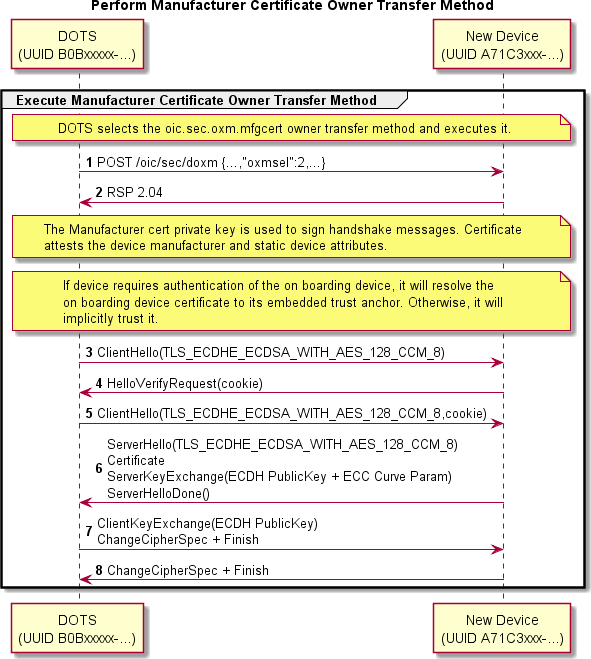
Figure 18 – Manufacturer Certificate Based OTM Sequence

Table 4 – Manufacturer Certificate Based OTM Details

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1, 2 | The DOTS notifies the Device that it selected the "Manufacturer Certificate" method. |
| 3 - 8 | A DTLS session is established using the device’s manufacturer certificate. The device’s manufacturer certificate may contain data attesting to the Device hardening and security properties. |

Если на шаге 1 выбран OTM на основе сертификата производителя, применяются следующие требования:

- На шаге 6:

- Новое Устройство должно использовать набор шифров DTLS, поддерживаемый для использования с OTM на основе сертификата производителя (см. Пункт 11.3.2.3),

- Новое Устройство не должно отправлять сообщение CertificateRequest.

ПРИМЕЧАНИЕ. Сообщение CertificateRequest отправляется при установке соединения DTLS для аутентификации устройства с использованием сертификатов (пункт 10.4.1).

#### 7.3.6.5 Соображения безопасности

Закрытый ключ сертификата производителя встроен в платформу с достаточной степенью уверенности в том, что закрытый ключ не может быть скомпрометирован.

Производитель платформы выдает сертификат производителя и подтверждает механизм защиты закрытого ключа.

### 7.3.7 OTM для конкретных поставщиков

#### 7.3.7.1 Общие положения OTM для конкретных поставщиков

OCF предвидит ситуации, когда поставщику потребуется внедрить OTM, учитывающую производственные ограничения или ограничения устройства. Ресурс Device OTM может быть расширен для этой цели.

OTM, зависящие от поставщика, должны придерживаться набора соглашений, которым следуют все OTM.

- OBT должен определить, какие типы учетных данных поддерживаются устройством. Это достигается путем запроса ресурса устройства «/oic/sec/doxm» для определения поддерживаемых типов учетных данных.

- OBT снабжает Устройство OC.

- OBT предоставляет UUID устройства и учетные данные для последующего доступа к OBT.

- OBT предоставит настройки второго оператора, достаточные для доступа к OCF Security владельца.

Домен после установления права собственности.

- OBT может выполнять дополнительные шаги по обеспечению, но не должен делать недействительными задачи по обеспечению, которые должны выполняться службой безопасности.

#### 7.3.7.2 Пример последовательности передачи владельца для конкретного поставщика

Пример последовательности OTM для конкретного поставщика показан на рисунке 19, а шаги описаны в таблице 5.

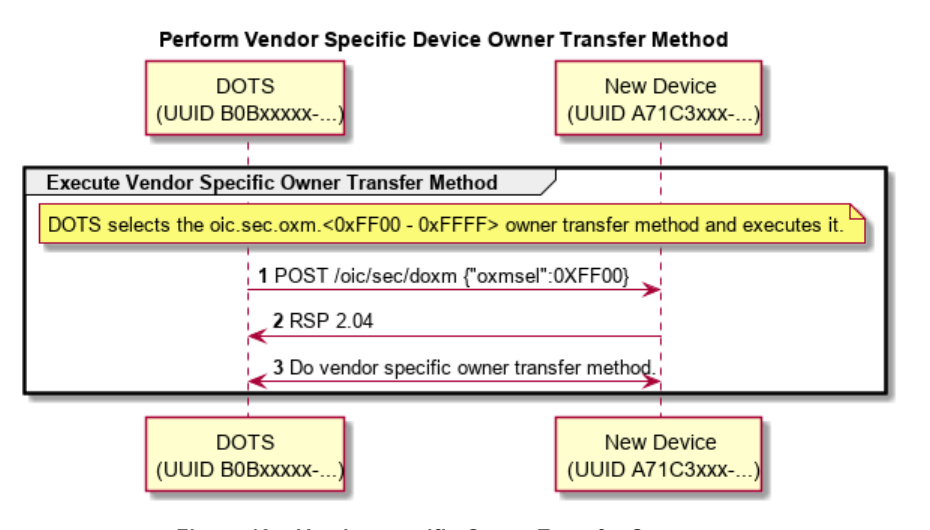


Рисунок 19 - Последовательность смены владельца, зависящая от поставщика

Table 5 – Vendor-specific Owner Transfer Details

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1, 2 | The DOTS selects a vendor-specific OTM. |
| 3 | The vendor-specific OTM is applied |

#### 7.3.7.3 Соображения безопасности

Поставщик несет ответственность за рассмотрение угроз безопасности и стратегии их устранения.

### 7.3.8 Установление полномочий владельца

После аутентификации OBT и нового устройства и установления зашифрованного соединения с использованием одного из определенных методов OTM могут быть предоставлены учетные данные владельца.

Учетные данные владельца предоставляются как часть метода передачи права собственности и могут быть предоставлены непосредственно CMS.

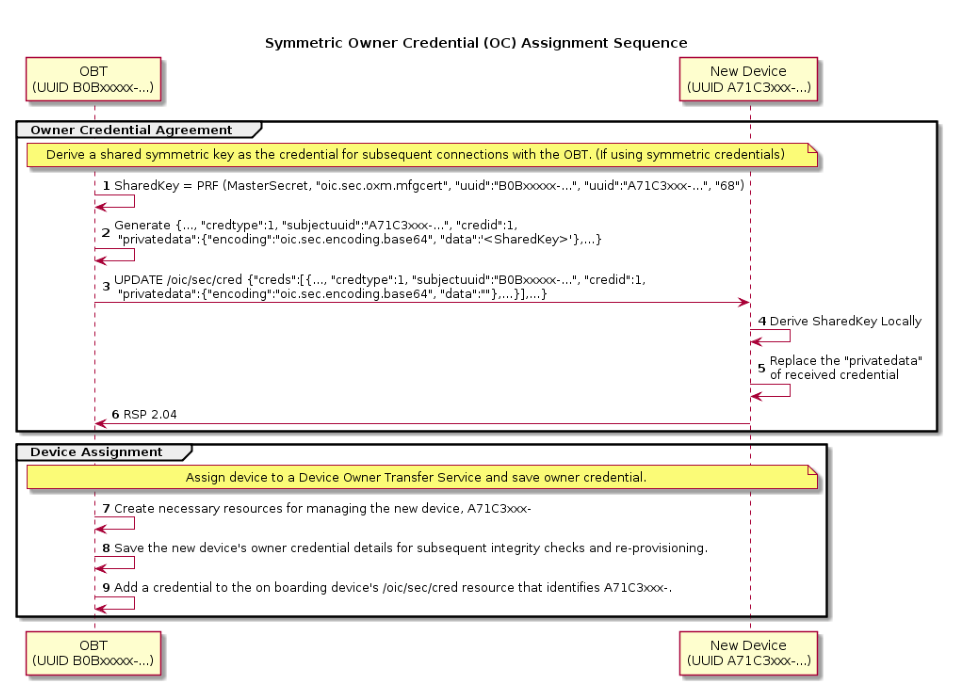
Шаги для установки учетных данных владельца устройства (OC) в рамках OTM:

1) OBT устанавливает UUID устройства и идентификатор владельца устройства.

2) OBT затем устанавливает симметричный OC устройства - см. Рисунок 20 и таблицу 6.

3) Настройте службы устройства.

4) Настройте устройство для однорангового взаимодействия.



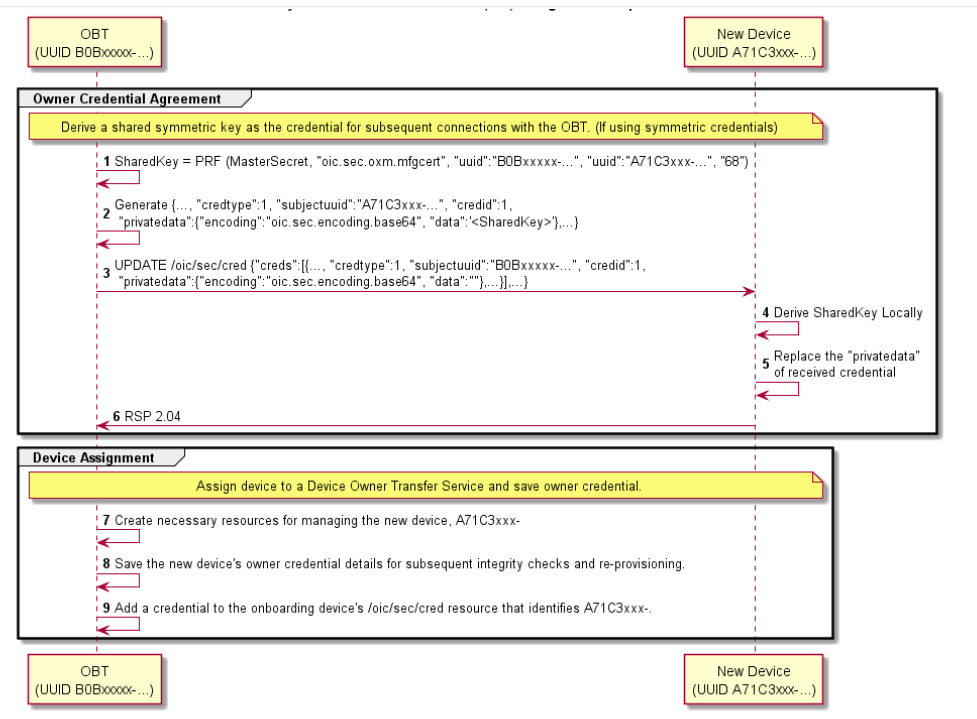


Figure 20 – Symmetric Owner Credential Provisioning Sequence

Table 6 – Symmetric Owner Credential Assignment Details

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1, 2 | The OBT uses a pseudo-random-function (PRF), the master secret resulting from the DTLS handshake, and other information to generate a symmetric key credential Resource Property - SharedKey. |
| 3 | The OBT creates a credential Resource Property set based on SharedKey and then sends the Resource Property set to the new Device with empty "privatedata" Property value. |
| 4, 5 | The new Device locally generates the SharedKey and updates it to the "privatedata" Property of the credential Resource Property set. |
| 6 | The new Device sends a success message. |
| 7 | The onboarding service creates a subjects Resource for the new device (e.g./A71C3xxx-…) |
| 8 | The onboarding service provisions its "/oic/svc/dots/subjects/A71C3xxx-/cred" Resource with the owner credential. Credential type is SYMMETRIC KEY. |
| 9 | (optional) The onboarding service provisions its own "/oic/sec/cred" Resource with the owner credential for new device. Credential type is SYMMETRIC KEY. |

В частности, когда OBT устанавливает симметричные учетные данные владельца как часть последовательности OTM:

- OBT генерирует общий ключ, используя метод вычисления учетных данных SharedKey, описанный в п. 7.3.2.

- OBT отправляет пустой ключ на ресурс «/ oic / sec / cred» нового Устройства, идентифицированный как симметричный парный ключ. UUID субъекта записи "/ oic / sec / cred" должен совпадать с UUID устройства OBT.

- После получения симметричных учетных данных владельца OBT новое Устройство должно независимо сгенерировать Общий ключ, используя метод расчета учетных данных SharedKey, описанный в 7.3.2, и сохранить его с учетными данными владельца.

- Новое Устройство должно использовать учетные данные владельца общего ключа, хранящиеся через Ресурс «/ oic / sec / cred», для аутентификации владельца во время последующих подключений.

### 7.3.9 Назначение профиля безопасности

Устройства OCF могли быть оценены в соответствии с профилем безопасности OCF. Результаты оценки могут быть доступны из сертификата производителя, веб-сервера OCF или другого общедоступного репозитория.

DOTS просматривает результаты оценки, чтобы определить, какие профили безопасности OCF разрешено владеть Устройству OCF, и настраивает Устройство с подмножеством оцененных профилей безопасности, наиболее подходящих для предполагаемой стратегии сегментации владельца домена безопасности OCF.

Поставщик устройства OCF должен установить значение производителя по умолчанию для свойства «supportedprofiles» ресурса «/ oic / sec / sp», чтобы оно соответствовало утвержденным в процессе тестирования и сертификации OCF.

Свойство "currentprofile" ресурса "/ oic / sec / sp" должно быть установлено в одно из значений, содержащихся в "supportedprofiles". Значение по умолчанию производителя должно быть повторно подтверждено, когда Устройство переходит в состояние СБРОС.

Устройство OCF должно разрешать обновление ресурса «/ oic / sec / sp» только тогда, когда Устройство находится в одном из следующих состояний устройства: RFOTM, RFPRO, SRESET, и может не разрешать какие-либо обновления в соответствии с указаниями профиля безопасности.

DOTS может обновить свойство «supportedprofiles» ресурса «/ oic / sec / sp» с помощью подмножества значений профилей безопасности OCF, полученных устройством в рамках тестирования соответствия OCF. DOTS может найти результаты соответствия, проверив сертификаты производителя, поставляемые с устройством OCF, выбрав свойство «credusage» ресурса «/ oic / sec / cred», имеющего значение «oic.sec.cred.mfgcert». DOTS может дополнительно найти результаты соответствия, посетив хорошо известный URI веб-сайта OCF, соответствующий полям расширения ocfCPLAttributes (пункт 9.4.2.2.7).

DOTS может выбрать подмножество профилей безопасности (из тех, которые оцениваются тестированием на соответствие OCF) на основе локальной политики.

В рамках подключения (пока сеанс OTM активен) DOTS должна настроить записи ACE, чтобы разрешить доступ к DOTS после подключения.

Служба DOTS должна обновить свойство "currentprofile" ресурса "/ oic / sec / sp" до значения, которое наиболее точно отражает предполагаемую стратегию развертывания Устройства владельцем домена безопасности OCF.

CMS может выдавать учетные данные роли, используя значение профиля безопасности (например, «sp-blue-v0 OID»), чтобы указать намерение владельца домена безопасности OCF сегментировать домен безопасности OCF в соответствии с профилем безопасности. CMS извлекает свойство supportedprofiles ресурса «/ oic / sec / sp» для выбора имен ролей, подтвержденных профилями безопасности, поддерживаемыми устройством, при выдаче учетных данных роли.

Если CMS выдает учетные данные роли на основе профиля безопасности, AMS предоставляет записи управления доступом, которые включают обозначения ролей.

## 7.4 Подготовка

### 7.4.1 Потоки обеспечения

#### 7.4.1.1 Общие положения о потоках подготовки

В рамках подключения нового устройства создается защищенный канал между новым устройством и OBT. После того, как статус владения устройством будет изменен на «принадлежащий», есть возможность начать подготовку. OBT предоставляет услуги поддержки, которые впоследствии следует использовать для завершения подготовки Устройства и текущего управления Устройством.

Устройство использует стратегию инициализации, ориентированную на клиента. Ресурс «/ oic / sec / pstat» определяет стратегию предоставления и текущий статус предоставления. Служба обеспечения должна определять, какая стратегия обеспечения является наиболее подходящей для домена безопасности OCF. См. 13.8 для получения дополнительных сведений.

#### 7.4.1.2 Предоставление, управляемое клиентом

Обеспечение, управляемое клиентом, основывается на сервисе обеспечения, который определяет серверы, которые нуждаются в

Provisioning затем выполняет все необходимые обязанности по обеспечению.

Пример инициализации, управляемой клиентом, показан на рисунке 21, а шаги описаны в таблице 7.

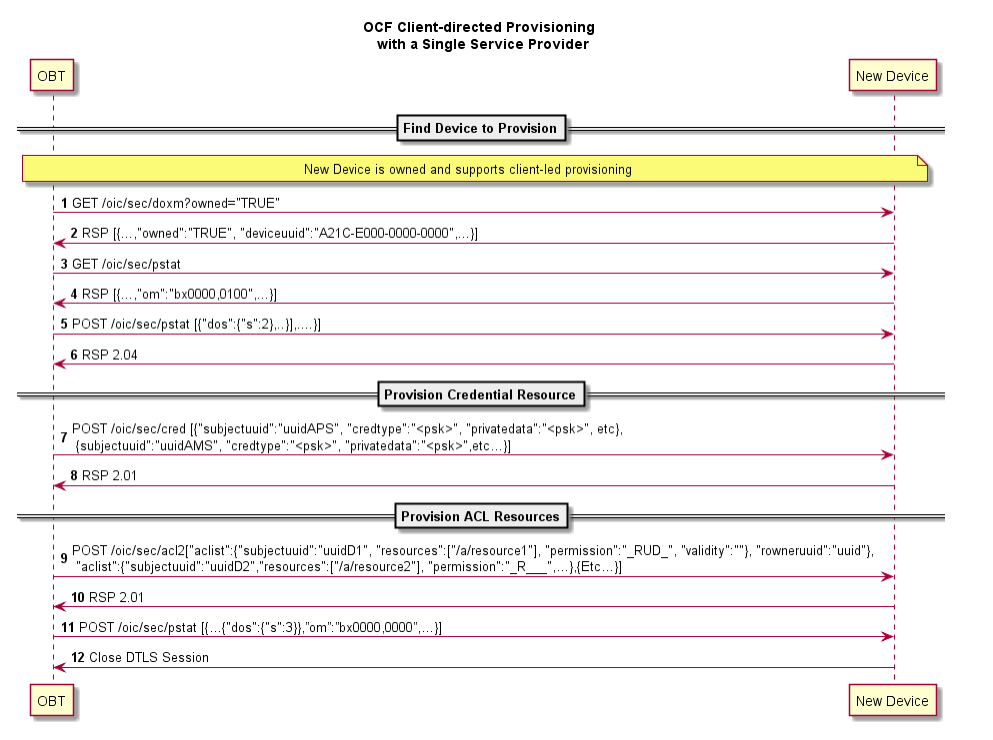


Рисунок 21 - Пример инициализации, управляемой клиентом

Таблица 7 - Шаги, описывающие инициализацию, управляемую клиентом

|  |  |
| --- | --- |
| Step | Description |
| 1 | Discover Devices that are owned and support Client- directed provisioning. |
| 2 | The "/oic/sec/doxm" Resource identifies the Device and it’s owned status. |
| 3 | DOTS (on OBT) obtains the new Device’s provisioning status found in "/oic/sec/pstat" Resource |
| 4 | The "pstat" Resource describes the types of provisioning modes supported and which is currently configured. A Device manufacturer should set a default current operational mode ("om"). If the "om" isn’t configured for Client-directed provisioning, its "om" value can be  changed. |
| 5 - 6 | Change Device state to RFPRO. |
| 7 - 8 | CMS (on OBT)instantiates the "/oic/sec/cred" Resource. It contains credentials for the provisioned services and other Devices |
| 9 - 10 | AMS (on OBT) instantiates "/oic/sec/acl2" Resource. |
| 11 | The new Device provisioning status mode is updated to reflect that ACLs have been configured. (RFNOP). |
| 12 | The secure session is closed. |

#### 7.4.1.3 Серверная инициализация [УСТАРЕЛО]

Этот пункт намеренно оставлен пустым.

#### 7.4.1.4 Серверная инициализация с использованием нескольких служб поддержки

[УСТАРЕЛО]

Этот пункт намеренно оставлен пустым.

# 8 Определения состояний подключения устройства

## 8.1 Общие сведения о подключении устройства

Как объяснено в п. 5.3, процесс подключения завершается после того, как право собственности на Устройство было передано и Устройству была предоставлена ​​соответствующая конфигурация / услуги, как описано в п. 5.4. На рисунке 22 показаны различные состояния, в которых устройство может находиться в течение жизненного цикла устройства. Устройство должно отклонять любые запросы на выполнение перехода между состояниями, не показанного на рисунке 22.

Свойство «/pstat.dos.s» является RW владельцем ресурса «/ oic / sec / pstat» (например, сервисом «doxs»), чтобы владелец ресурса мог удаленно обновлять состояние устройства. Когда Устройство находится в RFNOP или RFPRO, ACL могут использоваться, чтобы разрешить удаленное управление состоянием устройства другими устройствами. Когда состояние устройства - SRESET, OC устройства может быть единственным показателем авторизации для доступа к устройству. Владелец устройства может выполнять низкоуровневые проверки согласованности и повторную инициализацию, чтобы устройство подходило для перехода на RFPRO.

Рисунок 22 - Модель состояния устройства

Как показано на схеме, по завершении этапа инициализации Устройство входит в RFNOP, где у него есть все необходимое для начала взаимодействия с другими Устройствами. Пункт 8.5 определяет минимальную обязательную конфигурацию, которую должно содержать Устройство, чтобы считаться RFNOP.

В случае потери питания или сбоя Устройства Устройство должно оставаться в том же состоянии, в котором оно находилось до потери / отказа питания. Если владелец устройства или ресурса НАБЛЮДАЕТ "/pstat.dos.s", то переход к SRESET будет раннее оповещение об устройствах, которым может потребоваться проверка согласованности SVR.

Для того, чтобы подключение функционировало, на Устройстве должны быть установлены следующие Ресурсы:

1) Ресурс "/ oic / sec / doxm"

2) Ресурс "/ oic / sec / pstat"

3) Ресурс "/ oic / sec / cred"

Значения, содержащиеся в этих Ресурсах, указаны в определениях состояний в 8.2, 8.3, 8.4, 8.5 и 8.6. Также описывается политика доступа для этих и других SVR.

## 8.2 Определение состояния сброса устройства

/Pstat.dos.s = RESET определяется как «полный» сброс к заводским настройкам по умолчанию. Аппаратный сброс также определяет состояние, в котором актив устройства готов к передаче другому лицу.

Производитель платформы должен предоставить физический механизм (например, кнопку), который вызывает сброс платформы. Все Устройства, размещенные на одной Платформе, переводят свои состояния Устройства в состояние СБРОС, когда утверждается сброс Платформы.

Следующие ресурсы и их особые свойства должны иметь указанные значения:

- Свойство "own" ресурса "/ oic / sec / doxm" должно перейти в FALSE.

- Свойство "devowneruuid" ресурса "/ oic / sec / doxm" должно быть нулевым UUID.

- Свойство "deviceuuid" ресурса "/ oic / sec / doxm" должно быть установлено на значение производителя по умолчанию.

- Свойство "sct" ресурса "/ oic / sec / doxm" должно быть сброшено до значения по умолчанию, установленного производителем.

- Свойство "oxmsel" ресурса "/ oic / sec / doxm" должно быть сброшено до значения по умолчанию производителя.

- Свойство "isop" ресурса "/ oic / sec / pstat" должно иметь значение FALSE.

- Свойство "dos" ресурса "/ oic / sec / pstat" должно быть обновлено: dos.s должно быть равно "RESET".

- Свойство «om» (режимы работы) ресурса «/ oic / sec / pstat» должно быть установлено на значение по умолчанию производителя.

- Свойство «sm» (поддерживаемые режимы работы) ресурса «/ oic / sec / pstat» должно быть установлено на значение по умолчанию производителя.

- Свойство "creds" ресурса "/ oic / sec / cred" должно быть установлено по умолчанию производителя.

значение.

- Свойство "aclist2" ресурса "/ oic / sec / acl2" должно быть установлено на заводское значение по умолчанию.

значение.

- Свойство «rowneruuid» ресурсов «/ oic / sec / pstat», «/ oic / sec / doxm», «/ oic / sec / acl2» и «/ oic / sec / cred» должно быть нулевым UUID.

- Свойство "usedspace" ресурса "/ oic / sec / ael" должно быть установлено в 0.

- Свойство "categoryfilter" ресурса "/ oic / sec / ael" должно быть установлено на значение производителя по умолчанию.

- Свойство "priorityfilter" ресурса "/ oic / sec / ael" должно быть установлено на значение по умолчанию производителя.

- Свойство "events" ресурса "/ oic / sec / ael" должно быть установлено в пустой массив.

- Свойство "supportedprofiles" ресурса "/ oic / sec / sp" должно быть установлено на значение производителя по умолчанию.

- Свойство "currentprofile" ресурса "/ oic / sec / sp" должно быть установлено на значение производителя по умолчанию.

- Если Ресурс "/ oic / sec / sdi" предоставляется Устройством:

- Свойство "uuid" ресурса должно быть установлено равным нулю UUID.

- Свойство "name" Ресурса должно быть установлено в пустую строку.

- Свойство "priv" ресурса должно быть установлено на ЛОЖЬ.

- Устройство не принимает ни попытки подключения DTLS, ни попытки подключения TLS, ни какие-либо

другие запросы, включая запросы на обнаружение.

- Любые существующие соединения DTLS или TLS должны быть закрыты.

## 8.3 Устройство готово для определения состояния механизма передачи владельцу

Следующие Ресурсы и их особые свойства должны иметь значение, указанное, когда Устройство входит в готовность к передаче права собственности:

- Свойство "own" ресурса "/ oic / sec / doxm" должно иметь значение FALSE и перейдет в TRUE.

- Свойство "devowneruuid" ресурса "/ oic / sec / doxm" должно быть нулевым UUID.

- Свойство "deviceuuid" ресурса "/ oic / sec / doxm" должно быть установлено на значение производителя по умолчанию.

- Свойство "isop" ресурса "/ oic / sec / pstat" должно иметь значение FALSE.

- «dos» ресурса «/ oic / sec / pstat» должно быть обновлено: «dos.s» должно быть равно «RFOTM».

- Ресурс "/ oic / sec / cred" должен содержать учетные данные, если этого требует выбранный OTM.

- Если нет открытого подключения для подключения устройства, то

- Устройство должно предоставлять незащищенную конечную точку OCF для ресурсов «/ oic / sec / doxm» и «/ oic / sec / pstat».

- Для всех SVR, кроме «/ oic / sec / doxm» и «/ oic / sec / pstat»:

- SVR не должен открывать незащищенную конечную точку OCF.

- Запросы анонимного извлечения и обновлений (поступающие по неаутентифицированному каналу, например CoAP) для ресурса «/ oic / sec / doxm» должны быть удовлетворены.

- Если анонимный запрос на обновление ресурса «/ oic / sec / doxm» пытается обновить «oxmsel» до значения, которое не указано как поддерживаемое Устройством в «oxms», тогда Устройство должно отклонить запрос с соответствующим сообщение об ошибке (например, неверный запрос).

- Все запросы на получение к ресурсу «/ oic / sec / pstat» должны быть удовлетворены.

- Все другие запросы, за исключением запросов на получение к ресурсам обнаружения («/ oic / res», «/ oic / d» и «/ oic / p»), должны быть отклонены с соответствующим сообщением об ошибке (например, запрещено). .

- До успешного анонимного обновления «oxmsel» в «/ oic / sec / doxm» все попытки установить новые соединения DTLS должны быть отклонены.

- После успешного анонимного обновления «oxmsel» в «/ oic / sec / doxm»,

- Устройство должно позволять устанавливать подключение для подключения устройства (DOC), соответствующее свойству «oxmsel» ресурса «/ oic / sec / doxm» (как указано в пункте 7.3), и должно отклонять попытки установить другие подключения DTLS.

- Если есть открытый ДОК, то

- Для всех SVR:

- Устройство не должно открывать незащищенную конечную точку OCF для SVR.

- Все запросы, полученные через DOC, которые должны удовлетворять целевым DCR, независимо от конфигурации ACE в ресурсе "/ oic / sec / acl2".

- Все одноадресные запросы, которые не получены через открытый DOC устройства, должны быть отклонены с соответствующим сообщением об ошибке (например, запрещено), независимо от конфигурации ACE в ресурсе "/ oic / sec / acl2".

- Все попытки установить новые соединения DTLS должны быть отклонены.

- Если DOC закрыт в RFOTM, то Устройство перейдет в состояние СБРОС.

## 8.4 Устройство готово для определения состояния обеспечения

Следующие ресурсы и их особые свойства должны иметь значение, указанное, когда Устройство входит в готовность к предоставлению:

- Свойство «принадлежащее» ресурса «/ oic / sec / doxm» должно иметь значение ИСТИНА.

- Свойство "devowneruuid" ресурса "/ oic / sec / doxm" не должно быть нулевым UUID.

- Свойство "deviceuuid" ресурса "/ oic / sec / doxm" не должно быть нулевым UUID и должно быть установлено на значение, которое было определено во время обработки RFOTM.

- Свойство "oxmsel" ресурса "/ oic / sec / doxm" должно иметь значение фактического OTM, использованного во время передачи права собственности.

- Свойство "isop" ресурса "/ oic / sec / pstat" должно иметь значение FALSE.

- «dos» ресурса «/ oic / sec / pstat» должно быть обновлено: «dos.s» должно быть равно «RFPRO».

- Свойство "rowneruuid" каждого установленного Ресурса должно быть установлено на действительного владельца Ресурса (то есть объект, который уполномочен создавать или обновлять данный Ресурс). Неспособность установить

"rowneruuid" может привести к тому, что Ресурс потеряет свою актуальность.

- Ресурс «/ oic / sec / cred» должен содержать учетные данные для каждого объекта, на который ссылаются свойства «rowneruuid» и «devowneruuid».

- Все запросы к ресурсу «/ oic / sec / roles», полученные через соединение с взаимной аутентификацией, установленное с использованием сертификата идентичности, должны быть удовлетворены, независимо от конфигурация ACE в ресурсе "/ oic / sec / acl2" в соответствии с условиями пункта 10.4.2.

- Если есть открытый DOC, то все запросы, полученные через DOC, которые нацелены на DCR, должны быть удовлетворены, независимо от конфигурации ACE в ресурсе "/ oic / sec / acl2".

- Устройство должно позволять устанавливать соединения DTLS, аутентифицированные с помощью локально выданных учетных данных (пункты 10.2 и 10.4), и должно отклонять попытки установить другие соединения DTLS.

- Для всех SVR:

- SVR не должен открывать незащищенную конечную точку OCF.

- Устройство должно игнорировать все ACE с "subject", совпадающим с {"conntype": "anon-clear"} или {"conntype": "auth-crypt"} при принятии решений о доступе для запросов к SVR.

## 8.5 Определение состояния устройства, готового к нормальной работе

Следующие Ресурсы и их особые свойства должны иметь значение, указанное, когда Устройство входит в готовность к нормальной работе:

- Свойство «принадлежащее» ресурса «/ oic / sec / doxm» должно иметь значение ИСТИНА.

- Свойство "devowneruuid" ресурса "/ oic / sec / doxm" не должно быть нулевым UUID.

- Свойство "deviceuuid" ресурса "/ oic / sec / doxm" не должно быть нулевым UUID и должно быть установлено на ID, который был сконфигурирован во время OTM. Также значение свойства «di» в «/ oic / d» должно быть таким же, как и у deviceuuid.

- Свойство "oxmsel" ресурса "/ oic / sec / doxm" должно иметь значение фактического OTM, использованного во время передачи права собственности.

- Свойство "isop" ресурса "/ oic / sec / pstat" должно быть установлено на ИСТИНА сервером один раз.

в остальном переход на РФНОП завершен.

- «dos» ресурса «/ oic / sec / pstat» должно быть обновлено: «dos.s» должно быть равно «RFNOP».

- Свойство "rowneruuid" каждого установленного Ресурса должно быть установлено на действительного владельца Ресурса (то есть объект, который уполномочен создавать или обновлять данный Ресурс). Неспособность установить

"rowneruuid" приводит к потерянному Ресурсу.

- Ресурс «/ oic / sec / cred» должен содержать учетные данные для каждой службы, на которую ссылаются свойства «rowneruuid» и «devowneruuid».

- Все запросы к Ресурсу «/ oic / sec / roles», полученные через соединение с взаимной аутентификацией, установленное с использованием сертификата идентичности, должны быть удовлетворены, независимо от конфигурации ACE в Ресурсе «/ oic / sec / acl2», субъект к условиям п. 10.4.2.

- Если имеется открытый DOC, то для запросов, полученных через DOC, решения о доступе должны определяться следующим образом:

- Запрос, нацеленный на DCR, должен быть удовлетворен, независимо от конфигурации ACE.

в ресурсе "/ oic / sec / acl2".

- Запрос, нацеленный на NCR, должен быть удовлетворен путем сопоставления ACE в соответствии с обычной авторизацией запроса с «субъектом», совпадающим с типом соединения «нечеткое».

- Устройство должно позволять устанавливать соединения DTLS, аутентифицированные с использованием локально выданных учетных данных, и должно отклонять попытки установить другие соединения DTLS.

-

Для всех SVR:

- SVR не должен открывать незащищенную конечную точку OCF.

- Устройство должно игнорировать все ACE с "subject", совпадающим с {"conntype": "anon-clear"}

или {"conntype": "auth-crypt"} при принятии решений о доступе для запросов к SVR.

## 8.6 Определение состояния мягкого сброса устройства

Состояние мягкого сброса определяется (например, "/pstat.dos.s" = SRESET), где вход в это состояние означает

Устройство не работает, но остается в собственности текущего владельца. Устройство может выйти из SRESET путем аутентификации в DOTS (например, «rt» = «oic.r.doxs») с использованием OC, предоставленного во время первоначального подключения (но не должно требовать использования OTM /doxm.oxms).

Если учетные данные DOTS не могут быть найдены или определены как поврежденные, состояние устройства переходит в состояние СБРОС. Устройство должно оставаться в режиме SRESET, если учетные данные DOTS не могут подтвердить DOTS. Это предотвращает атаки типа «отказ в обслуживании», которые могут быть предприняты устройствами, не относящимися к DOTS.

В режиме SRESET следующие ресурсы и их особые свойства должны иметь указанные значения.

- Свойство «принадлежащее» ресурса «/ oic / sec / doxm» должно иметь значение ИСТИНА.

- Свойство "devowneruuid" ресурса "/ oic / sec / doxm" должно оставаться ненулевым.

- Свойство "deviceuuid" ресурса "/ oic / sec / doxm" должно оставаться ненулевым.

- Свойство "sct" ресурса "/ oic / sec / doxm" должно сохранять свое значение.

- Свойство "oxmsel" ресурса "/ oic / sec / doxm" должно сохранять свое значение.

- Свойство "isop" ресурса "/ oic / sec / pstat" должно иметь значение FALSE.

- Свойство "/oic/sec/pstat.dos.s" должно иметь значение SRESET.

- Свойство «om» (режимы работы) ресурса «/ oic / sec / pstat» должно быть «управляемым клиентом режимом».

- Свойство «sm» (поддерживаемые режимы работы) ресурса «/ oic / sec / pstat» может быть обновлено.

владельцем устройства (также известный как DOTS).

Свойство rowneruuid ресурсов «/ oic / sec / pstat», «/ oic / sec / doxm», «/ oic / sec / acl2» и «/ oic / sec / cred» может быть сброшено владельцем устройства. (также известный как DOTS) и повторно подготовлен.

- Все запросы к ресурсу "/ oic / sec / roles", полученные через соединение с взаимной аутентификацией, есть blished с использованием сертификата идентичности, независимо от конфигурации ACE в ресурсе «/ oic / sec / acl2», в соответствии с условиями пункта 10.4.2.

- Если есть открытый DOC, то все запросы, полученные через DOC, которые нацелены на DCR, должны быть удовлетворены, независимо от конфигурации ACE в ресурсе "/ oic / sec / acl2".

- Устройство должно позволять устанавливать соединения DTLS, аутентифицированные с использованием локально выданных учетных данных, и должно отклонять попытки установить другие соединения DTLS.

- Для всех SVR:

- SVR не должен открывать незащищенную конечную точку OCF.

- Устройство должно игнорировать все ACE с "subject", совпадающим с {"conntype": "anon-clear"} или {"conntype": "auth-crypt"} при принятии решений о доступе для запросов к SVR.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | | |
|  |
|  |
|  |  | | |
|  |
|  |
|  | |
|  | |
|  | |
|  | |
|  | | |
|  | | |
|  | | |
|  | | |
|  | | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |

# 9 Управление учетными данными безопасности

## 9.1 Преамбула

В этом разделе представлен обзор типов учетных данных в OCF, а также детали использования учетных данных, предоставления и текущего управления.

## 9.2 Жизненный цикл учетных данных

### 9.2.1 Общие сведения о жизненном цикле учетных данных

Жизненный цикл учетных данных OCF состоит из следующих этапов: (1) создание, (2) удаление, (3) обновление и (4) отзыв.

### 9.2.2 Создание

CMS может предоставить учетные данные для ресурса учетных данных на устройстве. Устройство должно проверить, авторизована ли CMS, сопоставив свойство rowneruuid ресурса «/ oic / sec / cred» с DeviceID учетных данных, которые CMS использовала для установления безопасного соединения.

Ресурсы учетных данных, созданные с помощью CMS, могут включать специализированные протоколы и сообщения выдачи учетных данных. Это может включать использование инфраструктуры открытых ключей (PKI), такой как центр сертификации (CA), управление симметричными ключами, например центр распределения ключей (KDC), или как часть действия по обеспечению DOTS, CMS или AMS.

### 9.2.3 Удаление

CMS может удалить учетные данные из Ресурса учетных данных. Устройство (например, Устройство, на котором размещен Ресурс учетных данных) должно удалить Ресурсы учетных данных, срок действия которых истек.

Ресурс учетных данных с истекшим сроком действия может быть удален для управления памятью и пространством для хранения. Удаление в управлении ключами OCF эквивалентно приостановке учетных данных.

### 9.2.4 Обновить

Обновление учетных данных может быть выполнено до истечения срока его действия. CMS выполняет обновление учетных данных.

Ресурс «/ oic / sec / cred» поддерживает истечение срока действия с помощью свойства Period. Обновление учетных данных может применяться, когда срок действия учетных данных скоро истечет или превысит максимальный порог для зашифрованных байтов.

Метод обновления учетных данных определяет параметры, доступные при выполнении обновления ключа. Свойство Period сообщает, когда должен истечь срок действия учетных данных. Устройство может заранее получить новые учетные данные, используя метод обновления учетных данных, используя текущие учетные данные, срок действия которых еще не истек, для обновления существующих учетных данных. Если Устройство не имеет внутреннего источника времени, текущее время следует получать от CMS через равные промежутки времени.

Если для установленных учетных данных разрешено истекать срок действия, DOTS должна повторно подключиться к Устройству, чтобы повторно применить шаги передачи владельца устройства.

Все Устройства должны поддерживать как минимум один метод обновления учетных данных.

### 9.2.5 Отзыв

Учетные данные, выданные CMS, могут иметь возможность отзыва. В ситуациях, когда метод отзыва включает подготовку объекта отзыва, который идентифицирует учетные данные, которые должны быть отозваны до его обычного срока истечения, создается ресурс учетных данных, содержащий информацию об отзыве, которая заменяет первоначально выданные учетные данные. Срок действия объекта отзыва должен соответствовать сроку действия отозванных учетных данных, чтобы объект отзыва был очищен по истечении срока действия.

Концептуально разумно рассматривать отзыв, применяемый к учетным данным или Устройству. При отзыве устройства утверждается, что все учетные данные, связанные с отозванным Устройством, должны рассматриваться для отзыва. Отзыв устройства необходим, если Устройство потеряно, украдено или взломано. Удаление учетных данных на отозванном Устройстве может быть невозможным или ненадежным.

## 9.3 Типы учетных данных

### 9.3.1 Преамбула

Ресурс «/ oic / sec / cred» поддерживает свойство типа учетных данных, которое поддерживает несколько криптографических ключей и другую информацию, используемую для аутентификации и защиты данных. Поддерживаемые типы учетных данных включают симметричный парный ключ, групповой симметричный групповой ключ, асимметричный ключ подписи, асимметричный ключ подписи с сертификатом и общий секрет (то есть ПИН или пароль). Устройство всегда должно поддерживать симметричный парный ключ и асимметричный ключ подписи с типами учетных данных сертификата. Другие типы учетных данных не являются обязательными.

### 9.3.2 Парные симметричные ключевые учетные данные

CMS должен предоставить ровно одно другое попарное симметричное удостоверение для однорангового Устройства. CMS не должен хранить попарные симметричные ключи, которые он предоставляет управляемым устройствам.

Парные ключи могут быть установлены через специальные протоколы согласования ключей.

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит симметричный ключ.

Свойство «PublicData» может содержать маркер, зашифрованный для однорангового Устройства, содержащий парный ключ.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют, что «PrivateData» останется закрытым.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа к ресурсам «/oic/sec/cred», «/oic/sec/roles», «/oic/sec/csr» для предотвращения несанкционированных изменений.

### 9.3.3 Учетные данные группового симметричного ключа

Групповые ключи - это симметричные ключи, совместно используемые группой Устройств (3 или более). Ключи группы используются для эффективного обмена данными между участниками группы.

Групповые ключи не обеспечивают аутентификацию Устройств, а только устанавливают членство в группе.

CMS должен предоставить учетные данные симметричного ключа группы членам группы. CMS поддерживает членство в группах.

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит симметричный ключ.

Свойство PublicData может содержать имя группы.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют, что «PrivateData» останется закрытым.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа к ресурсам «/ oic / sec / cred», «/ oic / sec / roles», «/ oic / sec / csr» для предотвращения несанкционированных изменений.

### 9.3.4 Учетные данные ключа асимметричной аутентификации

#### 9.3.4.1 Учетные данные ключа асимметричной аутентификации Общие

Учетные данные ключа асимметричной аутентификации содержат либо пару открытого и закрытого ключей, либо только открытый ключ. Закрытый ключ используется для подписи запросов проверки подлинности устройства. Открытый ключ используется для проверки ответа на запрос аутентификации устройства.

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит закрытый ключ.

Свойство PublicData содержит открытый ключ.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют, что «PrivateData» останется закрытым.

Устройства должны генерировать асимметричные пары ключей аутентификации внутри, чтобы гарантировать, что закрытый ключ известен только Устройству. См. 9.3.4.2, когда необходимо транспортировать материал закрытого ключа между Устройствами.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа.

ресурсов "/ oic / sec / cred", "/ oic / sec / roles", "/ oic / sec / csr" для предотвращения несанкционированного доступа

модификации.

#### 9.3.4.2 Внешнее создание учетных данных ключа асимметричной аутентификации

Устройства должны использовать стандартные отраслевые методы высокой надежности при разрешении ключа вне устройства.

создание и подготовка пар. Использование таких пар ключей следует свести к минимуму, особенно если ключ

пара неизменна и не может быть изменена или заменена после подготовки.

При использовании в процессе адаптации эти пары ключей могут использоваться для подтверждения того, что Устройство обладает

заявленные производителем свойства в сертификате, чтобы убедить DOTS или пользователя принять

подключение устройства. См. 7.3.3 для OTM, который использует такой сертификат для аутентификации

Device, а затем подготавливает новые учетные данные OCF Security Domain для использования.

### 9.3.5 Учетные данные ключа шифрования с асимметричным ключом

Учетные данные асимметричного ключа шифрования (KEK) используются для упаковки симметричных ключей, когда

распространение или хранение ключа.

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит закрытый ключ.

Свойство PublicData содержит открытый ключ.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют

"PrivateData" остается закрытым.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа. ресурсов "/ oic / sec / cred", "/ oic / sec / roles", "/ oic / sec / csr" для предотвращения несанкционированного доступа модификации.

### 9.3.6 Учетные данные сертификата

Учетные данные сертификата - это асимметричные ключи, которые сопровождаются сертификатом, выданным CMS.

или внешний центр сертификации (CA).

Протокол регистрации сертификатов используется для получения сертификата и подтверждения владения.

Выданный сертификат хранится с ресурсом учетных данных с асимметричным ключом.

Другие объекты, полезные для управления жизненным циклом сертификата, такие как статус отзыва сертификата:

связанный с Ресурсом учетных данных.

Либо ресурс учетных данных с асимметричным ключом, либо учетные данные самозаверяющего сертификата используются для завершить проверку пути.

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит закрытый ключ.

Свойство PublicData содержит выданный сертификат.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют

PrivateData остается закрытой.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа. ресурсов "/ oic / sec / cred", "/ oic / sec / roles", "/ oic / sec / csr" для предотвращения несанкционированного доступа модификации.

### 9.3.7 Учетные данные пароля

Свойство "PrivateData" в ресурсе "/ oic / sec / cred" содержит ПИН-код, пароль и другие

значения, полезные для изменения и проверки пароля.

Свойство PublicData может содержать имя пользователя или учетной записи, если применимо.

Свойство «OptionalData» может содержать статус отзыва.

Разработчик устройства должен применять усиленные методы хранения ключей, которые гарантируют

"PrivateData" остается закрытым.

Разработчик устройства должен применять соответствующую целостность, конфиденциальность и защиту доступа.

ресурсов "/ oic / sec / cred", "/ oic / sec / roles", "/ oic / sec / csr" для предотвращения несанкционированного доступа

модификации.

### 9.3.8 Учетные данные для прямого обеспечения контекста безопасности OSCORE

Запись учетных данных с типом учетных данных 64 используется для прямой подготовки OSCORE Security.

Параметры контекста для использования в сквозной защите одноадресных сообщений.

Свойство "privatedata" записи учетных данных с типом учетных данных 64 в "/ oic / sec / cred"

Ресурс содержит главный ключ OSCORE.

Запись учетных данных с типом учетных данных 64 должна отображать конфигурацию OSCORE («oscore»).

Недвижимость, в которую входят:

- Свойство "senderid", содержащее параметр идентификатора отправителя OSCORE.

- Свойство "recipientid", содержащее параметр идентификатора получателя OSCORE.

- Свойство "ssn" содержит доступное только для чтения значение, используемое для хранения последовательности отправителя OSCORE.

Число.

ПРИМЕЧАНИЕ: значения «senderid» и «recipientid» должны быть в шестнадцатеричном формате в нижнем регистре с кодировкой «0x».

префикс опущен.

См. Раздел 16.2 для описания параметров OSCORE.

### 9.3.9 Учетные данные для простой безопасной многоадресной рассылки

Существует два разных типа учетных данных, используемых для предоставления параметров контекста безопасности OSCORE.

используется в Simple Secure Multicast (SSM): один для клиентского контекста SSM, идентифицированный с помощью

"credtype": "128"; и один для контекста сервера SSM, идентифицированного с помощью «credtype»: «256». В

Клиент группы SSM, контекст безопасности OSCORE клиента (контекст отправителя) получен из

подготовленный клиентский контекст SSM. На серверах группы SSM безопасность OSCORE сервера

Контекст (контекст получателя) получается из подготовленного контекста сервера SSM.

Для обоих этих типов учетных данных свойство "privatedata" записи учетных данных в

Ресурс "/ oic / sec / cred" содержит значение главного секрета OSCORE группы SSM,

который генерируется OBT.

Запись учетных данных клиентского контекста SSM должна предоставлять свойство конфигурации OSCORE ("oscore"),

который для этого типа учетных данных должен включать:

- Свойство "senderid", содержащее параметр идентификатора отправителя OSCORE.

- Это значение выбирается и предоставляется OBT.

- Свойство «desc», содержащее описание использования контекста безопасности.

- Это свойство содержит удобочитаемое описание, предназначенное для идентификации

соответствующая группа SSM, если домен безопасности содержит несколько групп SSM.

- Это значение выбирается и предоставляется OBT

- Свойство "ssn" содержит доступное только для чтения значение, используемое для хранения последовательности отправителя OSCORE.

Число.

ПРИМЕЧАНИЕ 1. Ожидается, что значение «senderid» будет в шестнадцатеричном формате в нижнем регистре с опущенным префиксом кодирования «0x».

Учетная запись SSM Server Context должна включать конфигурацию OSCORE («oscore»).

Имущество, которое включает:

- Свойство "recipientid", содержащее параметр "ИД получателя группы OSCORE".

- Это значение одинаково для всех серверов в группе SSM и совпадает со значением параметра

"senderid" клиентского контекста для группы SSM

- Это значение выбирается и предоставляется OBT

- Свойство «desc», содержащее описание использования контекста безопасности.

- Это свойство содержит удобочитаемое описание, предназначенное для идентификации

соответствующая группа SSM, если домен безопасности содержит несколько групп SSM.

- Это значение выбирается и предоставляется OBT

ПРИМЕЧАНИЕ 2: Ожидается, что значение "recipientid" будет в шестнадцатеричном формате в нижнем регистре с опущенным префиксом кодирования "0x".

См. Раздел 16.3.3 для описания параметров OSCORE, используемых в SSM.

## 9.4 Управление ключами на основе сертификатов

### 9.4.1 Обзор

Для достижения аутентификации и безопасности транспорта во время связи в OCF Security Domain могут использоваться сертификаты, содержащие открытые ключи взаимодействующих сторон и частные ключи. Сертификат и закрытый ключ могут быть выпущены локальным или удаленным центром сертификации (CA).

Формат сертификата OCF является подмножеством формата X.509, разрешены только алгоритм эллиптической кривой и формат кодирования PEM, большинство дополнительных полей в X.509 не поддерживаются, поэтому формат соответствует требованиям ограниченного устройства.

CMS управляет жизненным циклом сертификатов для выдаваемых сертификатов. DOTS назначает CMS устройству, когда оно только что подключено.

### 9.4.2 Профили цифровых сертификатов X.509

#### 9.4.2.1 Общий профиль цифрового сертификата

Формат сертификата OCF - это подмножество формата X.509 (версия 3 или выше), как определено в IETF RFC 5280.

В этом разделе разработан профиль для облегчения использования сертификатов X.509 в приложениях OCF для тех сообществ, которые хотят использовать технологию X.509. Подробно описан формат сертификата X.509 v3 с дополнительной информацией, касающейся формата и семантики конкретных расширений OCF. Также перечислены поддерживаемые стандартные расширения сертификатов.

Формат сертификата: профиль сертификата OCF является производным от IETF RFC 5280. Однако этот документ не поддерживает поля «IssuyUniqueID» и «subjectUniqueID», которые устарели и не должны использоваться в контексте OCF. Если эти поля присутствуют в сертификате,

совместимые объекты должны игнорировать их содержимое.

Кодирование сертификата: соответствующие организации должны использовать почту с улучшенной конфиденциальностью (PEM) для кодирования сертификатов.

Иерархия сертификатов и параметры шифрования. OCF поддерживает трехуровневую иерархию для своей инфраструктуры открытых ключей (т. Е. Корневой центр сертификации, промежуточный центр сертификации и сертификаты EE). Cas, аккредитованный OCF, ДОЛЖЕН использовать ключи криптографии на основе эллиптических кривых (ECC) (secp256r1 - OID: 1.2.840.10045.3.1.7) и использовать алгоритм ecdsaWithSHA256 (OID: 1.2.840.10045.4.3.2) для подписей сертификатов. Открытые ключи криптографии с эллиптической кривой должны кодироваться с использованием несжатых точек эллиптической кривой.

Следующие пункты определяют поддерживаемые стандартные и настраиваемые расширения для профиля сертификатов OCF.

#### 9.4.2.2 Профиль и поля сертификата

##### 9.4.2.2.1 Профиль сертификата корневого ЦС

В таблице 8 описаны поля X.509 v1, необходимые для сертификатов корневого центра сертификации. Таблица 8 - Поля X.509 v1 для сертификатов корневого центра сертификации

|  |  |
| --- | --- |
| **V1 Field** | **Value / Remarks** |
| signatureAlgorithm | ecdsa-with-SHA256 (OID: 1.2.840.10045.4.3.2) |
| Version | v3 (value is 2) |
| SerialNumber | SHALL be a positive integer, unique among all certificates issued by a given CA |
| Issuer | SHALL match the Subject field |
| Subject | SHALL match the Issuer field |
| notBefore | The time at which the Root CA Certificate was generated.  See [10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16)compliant validity field formatting. |
| notAfter | No stipulation for expiry date.  See [10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16)compliant validity field formatting. |
| Subject Public Key Info | id-ecPublicKey (OID: 1.2.840.10045.2.1) secp256r1 (OID:1.2.840.10045.3.1.7)  Elliptic Curve Cryptography public keys shall be  encoded using uncompressed Elliptic Curve points. |

[Table 9](#_bookmark154) describes X.509 v3 extensions required for Root CA Certificates.

Table 9 - X.509 v3 extensions for Root CA Certificates

|  |  |  |  |
| --- | --- | --- | --- |
| Extension | Required/Optional | Criticality | Value / Remarks |
| authorityKeyIdentifier | OPTIONAL | Non-critical | N/A |
| subjectKeyIdentifier | OPTIONAL | Non-critical | N/A |
| keyUsage | REQUIRED | Critical | keyCertSign (5) & cRLSign  (6) bits shall be enabled.  digitalSignature(0) bit may be enabled.  All other bits shall not be enabled. |
| basicConstraints | REQUIRED | Critical | cA = TRUE  pathLenConstraint = not present (unlimited) |

Intermediate CA Certificate Profile

[Table 10](#_bookmark155) describes X.509 v1 fields required for Intermediate CA Certificates.

Table 10 - X.509 v1 fields for Intermediate CA Certificates

|  |  |
| --- | --- |
| V1 Field | Value / Remarks |
| signatureAlgorithm | ecdsa-with-SHA256 (OID: 1.2.840.10045.4.3.2) |
| Version | v3 (value is 2) |
| SerialNumber | SHALL be a positive integer, unique among all certificates issued by Root CA |
| Issuer | SHALL match the Subject field of the issuing Root CA |
| Subject | (no stipulation) |
| notBefore | The time at which the Intermediate CA Certificate was generated.  See clause [10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16) compliant validity field formatting. |
| notAfter | No stipulation for expiry date.  See clause[10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16) compliant validity field formatting. |
| Subject Public Key Info | id-ecPublicKey (OID: 1.2.840.10045.2.1) secp256r1 (OID:1.2.840.10045.3.1.7)  Elliptic Curve Cryptography public keys shall be  encoded using uncompressed Elliptic Curve points. |

[Table 11](#_bookmark156) describes X.509 v3 extensions required for Intermediate CA Certificates.

Table 11 – X.509 v3 extensions for Intermediate CA Certificates

|  |  |  |  |
| --- | --- | --- | --- |
| Extension | Required/Optional | Criticality | Value / Remarks |
| authorityKeyIdentifier | OPTIONAL | Non-critical | N/A |
| subjectKeyIdentifier | OPTIONAL | Non-critical | N/A |
| keyUsage | REQUIRED | Critical | keyCertSign (5) & cRLSign  (6) bits shall be enabled.  digitalSignature (0) bit may be enabled  All other bits shall not be enabled. |
| basicConstraints | REQUIRED | Critical | cA = TRUE |
|  |  |  | pathLenConstraint = 0 (can only sign End-Entity certs) |
| certificatePolicies | OPTIONAL | Non-critical | (no stipulation) |
| cRLDistributionPoints | OPTIONAL | Non-critical | 1 or more URIs where the Certificate Revocation List (CRL) from the Root can be obtained. |
| authorityInformationAccess | OPTIONAL | Non-critical | OCSP URI – the URI of the Root CA’s OCSP Responder |

End-Entity Black Certificate Profile

[Table 12](#_bookmark157) describes X.509 v1 fields required for End-Entity Certificates used for Black security profile.

Table 12 – X.509 v1 fields for End-Entity Certificates

|  |  |
| --- | --- |
| V1 Field | Value / Remarks |
| signatureAlgorithm | ecdsa-with-SHA256 (OID: 1.2.840.10045.4.3.2) |
| Version | v3 (value is 2) |
| SerialNumber | SHALL be a positive integer, unique among all certificates issued by the Intermediate CA |
| Issuer | SHALL match the Subject field of the issuing Intermediate CA |
| Subject | Subject DN shall include:  o=OCF-verified device manufacturer organization name.  The Subject DN may include other attributes (e.g. cn, c, ou, etc.) with no stipulation by OCF. |
| notBefore | The time at which the End-Entity Certificate was generated.  See clause [10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16) compliant validity field formatting. |
| notAfter | No stipulation.  See clause [10.4.5](#_bookmark179) for details around [IETF RFC 5280-](#_bookmark16) compliant validity field formatting. |
| Subject Public Key Info | id-ecPublicKey (OID: 1.2.840.10045.2.1) secp256r1 (OID:1.2.840.10045.3.1.7)  Elliptic Curve Cryptography public keys shall be encoded using uncompressed Elliptic Curve points. |

[Table 13](#_bookmark158) describes X.509 v3 extensions required for End-Entity Certificates.

Table 13 – X.509 v3 extensions for End-Entity Certificates

|  |  |  |  |
| --- | --- | --- | --- |
| Extension | Required/ Optional | Criticality | Value / Remarks |
| authorityKeyIdentifier | OPTIONAL | Non-critical | N/A |
| subjectKeyIdentifier | OPTIONAL | Non-critical | N/A |
| keyUsage | REQUIRED | Critical | digitalSignature (0) and keyAgreement(4) bits SHALL be the only bits enabled |
| basicConstraints | OPTIONAL | Non-Critical | cA = FALSE |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | pathLenConstraint = not present |
| certificatePolicies | OPTIONAL | Non-critical | End-Entity certificates  chaining to an OCF Root CA SHOULD contain at least  one PolicyIdentifierId set to the OCF Certificate Policy OID –  (1.3.6.1.4.1.51414.0.1.2)  corresponding to the version of the OCF Certificate Policy under which it was issued.  Additional manufacturer-  specific CP OIDs may also be populated. |
| extendedKeyUsage |  |  | The following |
|  |  |  | extendedKeyUsage (EKU) |
|  |  |  | OIDs SHALL both be |
|  |  |  | present: |
|  |  |  | serverAuthentication - |
|  |  |  | 1.3.6.1.5.5.7.3.1 |
|  |  |  | clientAuthentication - |
|  |  |  | 1.3.6.1.5.5.7.3.2 |
|  | REQUIRED | Non-critical | Exactly ONE of the following OIDs SHALL be present: |
|  |  |  | Identity certificate - |
|  |  |  | 1.3.6.1.4.1.44924.1.6 |
|  |  |  | Role certificate - |
|  |  |  | 1.3.6.1.4.1.44924.1.7 |
|  |  |  | End-Entity certificates |
|  |  |  | SHALL NOT contain the |
|  |  |  | anyExtendedKeyUsage OID |
|  |  |  | (2.5.29.37.0) |
| subjectAlternativeName |  |  | The subjectAltName  extension is used to encode one or more Role ID values in role certificates, binding  the roles to the subject public key.  When the  extendedKeyUsage (EKU) extension contains the  Identity Certificate OID  (1.3.6.1.4.1.44924.1.6), the  subjectAltName extension SHOULD NOT be present. |
|  | REQUIRED UNDER  CERTAIN CONDITIONS | Non-critical | If the EKU extension  contains the Role Certificate OID (1.3.6.1.4.1.44924.1.7),  the subjectAltName  extension SHALL be present and populated as follows: Each GeneralName in the GeneralNames SEQUENCE which encodes a role shall  be a directoryName, which is of type Name. Name is an  X.501 Distinguished Name.  Each Name shall contain exactly one CN (Common  Name) component, and zero or one OU (Organizational Unit) components. The OU  component, if present, shall |
|  |  |  | specify the authority that  defined the semantics of the role. If the OU component is absent, the certificate issuer has defined the role. The CN component shall encode the role ID. Other GeneralName types in the SEQUENCE  may be present, but shall  not be interpreted as roles. The role, and authority shall be encoded as ASN.1  PrintableString type, the restricted character set [0- 9a-z-A-z '()+,-./:=?]. |
| cRLDistributionPoints | OPTIONAL | Non-critical | 1 or more URIs where the Certificate Revocation List (CRL) from the Intermediate CA can be obtained. |
| authorityInformationAccess | OPTIONAL | Non-critical | OCSP URI – the URI of the Intermediate CA’s OCSP Responder |
| OCF Compliance | OPTIONAL | Non-critical | See [9.4.2.2.4](#_bookmark159) |
| Manufacturer Usage Description (MUD) | OPTIONAL | Non-critical | Contains a single Uniform Resource Locator (URL) that points to an on-line  Manufacturer Usage  Description concerning the certificate subject. See  [9.4.2.2.5](#_bookmark160) |
| OCF Security Claims | OPTIONAL | Non-critical | Contains a list of security claims above those required by this OCF Compliance  version or Security Profile. See [9.4.2.2.6](#_bookmark161) |
| OCF CPL Attributes | OPTIONAL | Non-critical | Contains the list of OCF Attributes used to perform OCF Certified Product List lookups |

OCF Compliance X.509v3 Extension

|  |  |
| --- | --- |
| 2159 | The OCF Compliance Extension defines required parameters to correctly identify the type of Device, |
| 2160 | its manufacturer, its OCF Version, and the Security Profile compliance of the device. |
| 2161 | The extension carries an "ocfVersion" field which provides the specific base version of the OCF |
| 2162 | documents the device implements. The "ocfVersion" field shall contain a sequence of three integers |
| 2163 | ("major", "minor", and "build"). For example, if an entity is certified to be compliant with OCF |
| 2164 | specifications 1.3.2, then the "major", "minor", and "build" fields of the "ocfVersion" will be set to |
| 2165 | "1", "3", and "2" respectively. The "ocfVersion" may be used by Security Profiles to denote |
| 2166 | compliance to a specified base version of the OCF documents. |
| 2167 | The "securityProfile" field shall carry the ocfSecurityProfile OID(s) (clause [14.8.3](#_bookmark319)) of one or more |
| 2168 | supported Security Profiles associated with the certificate in string form (UTF-8). All Security |
| 2169 | Profiles associated with the certificate should be identified by this field. |
| 2170 | The extension shall also carry two string fields (UTF-8): "DeviceName" and "deviceManufacturer". |
| 2171 | The fields carry human-readable descriptions of the Device’s name and manufacturer, respectively. |
| 2172 | The ASN.1 definition of the OCFCompliance extension (OID – 1.3.6.1.4.1.51414.1.0) is defined as |
| 2173 | follows: |
| 2174 | id-OCF OBJECT IDENTIFIER ::= { iso(1) identified-organization(3) dod(6) internet(1) |

private(4) enterprise(1) OCF(51414) } id-ocfX509Extensions OBJECT IDENTIFIER ::= { id-OCF 1 }

id-ocfCompliance OBJECT IDENTIFIER ::= { id-ocfX509Extensions 0 }

ocfVersion ::= SEQUENCE { major INTEGER,

--Major version number minor INTEGER,

--Minor version number build INTEGER,

--Build/Micro version number

}

ocfCompliance ::= SEQUENCE {

version ocfVersion,

--Device/OCF version

securityProfile SEQUENCE SIZE (1..MAX) OF ocfSecurityProfileOID,

--Sequence of OCF Security Profile OID strings

--Clause [14.8.2](#_bookmark316) defines valid ocfSecurityProfileOIDs

deviceName UTF8String,

--Name of the device deviceManufacturer UTF8String,

--Human-Readable Manufacturer

--of the device

}

##### Manufacturer Usage Description (MUD) X.509v3 Extension

The goal of the Manufacturer Usage Description (MUD) extension is to provide a means for devices to signal to the network the access and network functionality they require to properly function. Access controls can be more easily achieved and deployed at scale when the MUD extension is used.

The MUD X.509 v3 extension is specified in [IETF RFC 8520](#_bookmark26) with the full ASN.1 definition in clause 11.

##### OCF Security Claims X.509v3 Extension

The OCF Security Claims Extension defines a list of OIDs representing security claims that the manufacturer/integrator is making as to the security posture of the device above those required by the OCF Compliance version or that of the OCF Security Profile being indicated by the device.

The purpose of this extension is to allow for programmatic evaluation of assertions made about security to enable some platforms/policies/administrators to better understand what is being onboarded or challenged.

The ASN.1 definition of the OCF Security Claims extension (OID – 1.3.6.1.4.1.51414.1.1) is defined as follows:

id-OCF OBJECT IDENTIFIER ::= { iso(1) identified-organization(3) dod(6) internet(1)

private(4) enterprise(1) OCF(51414) } id-ocfX509Extensions OBJECT IDENTIFIER ::= { id-OCF 1 }

id-ocfSecurityClaims OBJECT IDENTIFIER ::= { id-ocfX509Extensions 1 }

claim-secure-boot ::= ocfSecurityClaimsOID { id-ocfSecurityClaims 0 }

--Device claims that the boot process follows a procedure trusted

--by the firmware and the BIOS

claim-hw-backed-cred-storage ::= ocfSecurityClaimsOID { id-ocfSecurityClaims 1 }

2230

2231

2232

2233

2234

2235

2236

2237

2238

2239

2240

2241

2242

2243

2244

2245

2246

2247

2248

2249

2250

2251

2252

2253

2254

2255

2256

2257

2258

2259

2260

2261

2262

2263

2264

2265

2266

2267

2268

2269

2270

2271

2272

--Device claims that credentials are stored in a specialized hardware

--protection environment such as a Trusted Platform Module (TPM) or

--similar mechanism.

ocfSecurityClaimsOID ::= OBJECT IDENTIFIER

ocfSecurityClaims ::= SEQUENCE SIZE (1..MAX) of ocfSecurityClaimsOID

##### OCF Certified Product List Attributes X.509v3 Extension

The OCF Certified Product List Extension defines required parameters to utilize the OCF Compliance Management System Certified Product List (OCMS-CPL). This clause is only applicable if you plan to utilize the OCMS-CPL. The OBT may make use of these attributes to verify the compliance level of a device.

The extension carries the OCF CPL Attributes: IANA Private Enterprise Number (PEN), Model and Version.

The 'cpl-at-IANAPen' IANA Private Enterprise Number (PEN) provides the manufacturer's unique PEN established in the IANA PEN list located at: https://[www.iana.org/assignments/enterprise-](http://www.iana.org/assignments/enterprise-) numbers. The 'cpl-at-IANAPen' field found in end-products shall be the same information as reported during OCF Certification.

The 'cpl-at-model' represents an OCF-Certified product's model name. The 'cpl-at-model' field found in end-products shall be the same information as reported during OCF Certification.

The 'cpl-at-version' represents an OCF-Certified product's version. The 'cpl-at-version' field found in end-products shall be the same information as reported during OCF Certification.

The ASN.1 definition of the OCF CPL Attributes extension (OID – 1.3.6.1.4.1.51414.1.2) is defined as follows:

id-OCF OBJECT IDENTIFIER ::= { iso(1) identified-organization(3) dod(6) internet(1)

private(4) enterprise(1) OCF(51414) } id-ocfX509Extensions OBJECT IDENTIFIER ::= { id-OCF 1 }

id-ocfCPLAttributes OBJECT IDENTIFIER ::= { id-ocfX509Extensions 2 }

cpl-at-IANAPen ::= OBJECT IDENTIFIER { id-ocfCPLAttributes 0 } cpl-at-model ::= OBJECT IDENTIFIER { id-ocfCPLAttributes 1 } cpl-at-version ::= OBJECT IDENTIFIER { id-ocfCPLAttributes 2 }

ocfCPLAttributes ::= SEQUENCE {

cpl-at-IANAPen UTF8String,

--Manufacturer's registered IANA Private Enterprise Number cpl-at-model UTF8String,

--Device OCF Security Profile

cpl-at-version UTF8String

--Name of the device

|  |  |  |
| --- | --- | --- |
| 2273 | } |  |
| 2274 | **9.4.2.3** | **Supported Certificate Extensions** |
| 2275 | As these certificate extensions are a standard part of [IETF RFC 5280,](#_bookmark16) this document includes the | |
| 2276 | clause number from that RFC to include it by reference. Each extension is summarized here, and | |
| 2277 | any modifications to the RFC definition are listed. Devices MUST implement and understand the | |
| 2278 | extensions listed here; other extensions from the RFC are not included in this document and | |
| 2279 | therefore are not required. [10.4](#_bookmark173) describes what Devices must implement when validating certificate | |
| 2280 | chains, including processing of extensions, and actions to take when certain extensions are absent. | |

2281

2282

2283

2284

2285

2286

2287

2288

2289

2290

2291

2292

2293

2294

2295

2296

2297

2298

2299

2300

2301

2302

2303

2304

2305

2306

2307

2308

2309

2310

2311

2312

2313

2314

2315

2316

2317

2318

2319

2320

2321

2322

2323

2324

2325

2326

2327

2328

2329

2330

2331

2332

2333

2334

2335

– Authority Key Identifier (4.2.1.1)

The Authority Key Identifier (AKI) extension provides a means of identifying the public key corresponding to the private key used to sign a certificate. This document makes the following modifications to the referenced definition of this extension:

The "authorityCertIssuer" or "authorityCertSerialNumber" fields of the "AuthorityKeyIdentifier" sequence are not permitted; only "keyIdentifier" is allowed. This results in the following grammar definition:

id-ce-authorityKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 35 }

AuthorityKeyIdentifier ::= SEQUENCE {

keyIdentifier [0] KeyIdentifier }

KeyIdentifier ::= OCTET STRING

– Subject Key Identifier (4.2.1.2)

The Subject Key Identifier (SKI) extension provides a means of identifying certificates that contain a particular public key.

This document makes the following modification to the referenced definition of this extension:

Subject Key Identifiers SHOULD be derived from the public key contained in the certificate’s "SubjectPublicKeyInfo" field or a method that generates unique values. This document RECOMMENDS the 256-bit SHA-2 hash of the value of the BIT STRING "subjectPublicKey" (excluding the tag, length, and number of unused bits). Devices verifying certificate chains must not assume any particular method of computing key identifiers, however, and must only base matching AKI’s and SKI’s in certification path constructions on key identifiers seen in certificates.

– Subject Alternative Name

If the EKU extension is present, and has the value XXXXXX, indicating that this is a role certificate, the Subject Alternative Name (subjectAltName) extension shall be present and interpreted as described below. When no EKU is present, or has another value, the "subjectAltName" extension SHOULD be absent. The "subjectAltName" extension is used to encode one or more Role ID values in role certificates, binding the roles to the subject public key. The "subjectAltName" extension is defined in [IETF RFC 5280](#_bookmark16) (See 4.2.1.6):

id-ce-subjectAltName OBJECT IDENTIFIER ::= { id-ce 17 } SubjectAltName ::= GeneralNames

GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName

GeneralName ::= CHOICE {

otherName [0] OtherName,

rfc5322Name [1] IA5String,

dNSName [2] IA5String,

x400Address [3] ORAddress,

directoryName [4] Name,

ediPartyName [5] EDIPartyName,

uniformResourceIdentifier [6] IA5String, iPAddress [7] OCTET STRING,

registeredID [8] OBJECT IDENTIFIER }

EDIPartyName ::= SEQUENCE {

nameAssigner [0] DirectoryString OPTIONAL,

partyName [1] DirectoryString }

Each "GeneralName" in the "GeneralNames" SEQUENCE which encodes a role shall be a "directoryName", which is of type Name. Name is an X.501 Distinguished Name. Each Name shall contain exactly one CN (Common Name) component, and zero or one OU (Organizational Unit) components. The OU component, if present, shall specify the authority that defined the

2336

2337

2338

2339

2340

2341

2342

2343

2344

2345

2346

2347

2348

2349

2350

2351

2352

2353

2354

2355

2356

2357

2358

2359

2360

2361

2362

2363

2364

2365

2366

2367

2368

2369

2370

2371

2372

2373

2374

2375

2376

2377

2378

semantics of the role. If the OU component is absent, the certificate issuer has defined the role. The CN component shall encode the role ID. Other "GeneralName" types in the SEQUENCE may be present, but shall not be interpreted as roles. Therefore, if the certificate issuer includes non-role names in the "subjectAltName" extension, the extension should not be marked critical.

The role, and authority need to be encoded as ASN.1 "PrintableString" type, the restricted character set [0-9a-z-A-z '()+,-./:=?].

– Key Usage (4.2.1.3)

The key usage extension defines the purpose (e.g., encipherment, signature, certificate signing) of the key contained in the certificate. The usage restriction might be employed when a key that could be used for more than one operation is to be restricted.

This document does not modify the referenced definition of this extension.

* Basic Constraints (4.2.1.9)

The basic constraints extension identifies whether the subject of the certificate is a CA and the maximum depth of valid certification paths that include this certificate. Without this extension, a certificate cannot be an issuer of other certificates.

This document does not modify the referenced definition of this extension.

* Extended Key Usage (4.2.1.12)

Extended Key Usage describes allowed purposes for which the certified public key may can be used. When a Device receives a certificate, it determines the purpose based on the context of the interaction in which the certificate is presented, and verifies the certificate can be used for that purpose.

This document makes the following modifications to the referenced definition of this extension: CAs SHOULD mark this extension as critical.

CAs MUST NOT issue certificates with the anyExtendedKeyUsage OID (2.5.29.37.0).

The list of OCF-specific purposes and the assigned OIDs to represent them are:

– Identity certificate 1.3.6.1.4.1.44924.1.6

– Role certificate 1.3.6.1.4.1.44924.1.7

##### Cipher Suite for Authentication, Confidentiality and Integrity

OCF compliant entities shall support TLS version 1.2. Compliant entities shall support TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8 cipher suite as defined in IETF RFC 7251 and may support additional ciphers as defined in the TLS v1.2 specifications.

##### Encoding of Certificate

See [9.4.2](#_bookmark151) for details.

##### Certificate Revocation List (CRL) Profile [Deprecated]

This clause is intentionally left blank.

##### Resource Model

Device certificates and private keys are kept in "cred" Resource.

The "cred" Resource contains the certificate information pertaining to the Device. The "PublicData" Property holds the device certificate and CA certificate chain. "PrivateData" Property holds the Device private key paired to the certificate. (See [13.3](#_bookmark220) for additional detail regarding the "/oic/sec/cred" Resource).

2379

##### Certificate Provisioning

|  |  |
| --- | --- |
| 2380 | The CMS (e.g. a hub or a smart phone) issues certificates for new Devices. |
| 2381 | The CA in the CMS retrieves a Device’s public key and proof of possession of the private key, |
| 2382 | generates a Device’s certificate signed by this CA certificate, and then the CMS transfers them to |
| 2383 | the Device including its CA certificate chain. Optionally, the CMS can also transfer one or more |
| 2384 | role certificates, which shall have the format described in clause [9.4.2.](#_bookmark151) The "subjectPublicKey" of |
| 2385 | each role certificate shall match the "subjectPublicKey" in the Device certificate. |
| 2386 | In the sequence in [Figure 23,](#_bookmark167) the Certificate Signing Request (CSR) is defined by PKCS#10 in |
| 2387 | [IETF RFC 2986,](#_bookmark11) and is included here by reference. |
| 2388 | The sequence flow of a certificate transfer for a Client-directed model is described in [Figure 23](#_bookmark167). |
| 2389 | 1) The CMS retrieves a CSR from the Device that requests a certificate. In this CSR, the Device |
| 2390 | shall place its requested UUID into the subject and its public key in the "SubjectPublicKeyInfo". |
| 2391 | The Device determines the public key to present; this may be an already-provisioned key it has |
| 2392 | selected for use with authentication, or if none is present, it may generate a new key pair |
| 2393 | internally and provide the public part. The key pair shall be compatible with the allowed cipher |
| 2394 | suites listed in [9.4.2.4](#_bookmark163) and [11.3.4,](#_bookmark191) since the certificate will be restricted for use in OCF |
| 2395 | authentication. |
| 2396 | 2) 2) Alternatively, the CMS generates and provisions a private key and corresponding certificate |
| 2397 | directly to the Device. |
| 2398 | 3) The CMS transfers the issued certificate and CA chain to the designated Device using the same |
| 2399 | credid, to maintain the association with the private key. The credential type ("oic.sec.cred") |
| 2400 | used to transfer certificates in [Figure 23](#_bookmark167) is also used to transfer role certificates, by including |
| 2401 | multiple credentials in the POST from CMS to Device. Identity certificates shall be stored with |
| 2402 | the credusage Property set to "oic.sec.cred.cert" and role certificates shall be stored with the |
| 2403 | credusage Property set to "oic.sec.cred.rolecert". |

##### Generated by PlantUMLFigure 23 – Client-directed Certificate Transfer

* + 1. **CRL Provisioning [Deprecated]**

This clause is intentionally left blank.

# 10 Аутентификация устройства

## 10.1 Аутентификация устройства Общие

Когда Клиент обращается к ограниченному Ресурсу на Сервере, Сервер должен аутентифицировать Клиента. Клиенты должны аутентифицировать Серверы при запросе доступа. Клиенты также могут утверждать одну или несколько ролей, которые сервер может использовать при принятии решений по управлению доступом. Роли могут быть утверждены, когда аутентификация устройства выполняется с помощью сертификатов.

## 10.2 Аутентификация устройства с использованием учетных данных с симметричным ключом

При использовании симметричных ключей для аутентификации серверное устройство должно включать сообщение «ServerKeyExchange» и устанавливать «psk\_identity\_hint» на UUID серверного устройства. Клиент должен подтвердить, что у него есть учетные данные с UUID субъекта, установленным на UUID устройства Сервера, и типом учетных данных PSK. В противном случае Клиент должен ответить ошибкой unknown\_psk\_identity или другой подходящей ошибкой.

Если Клиент находит подходящие учетные данные PSK, он должен ответить сообщением «ClientKeyExchange», которое включает «psk\_identity», установленное для UUID устройства Клиента. Сервер должен проверить, что у него есть учетные данные с совпадающим UUID Субъекта и типом. В противном случае Сервер должен ответить «unknown\_psk\_identity» или другой подходящий код ошибки. Если это так, то он должен продолжить работу по протоколу DTLS, и и Клиент, и Сервер должны вычислить полученный в результате главный секретный ключ.

## 10.3 Аутентификация устройства с использованием необработанных учетных данных асимметричного ключа

При использовании необработанных асимметричных ключей для аутентификации Клиент и Сервер должны включать подходящий открытый ключ из учетных данных, привязанных к их Устройству. Каждое Устройство должно проверять, что предоставленный открытый ключ соответствует полю Public Data имеющегося у него удостоверения, и использовать соответствующий UUID Субъекта удостоверения для идентификации однорангового Устройства.

## 10.4 Аутентификация устройства с помощью сертификатов

10.4.1 Аутентификация устройства с помощью сертификатов Общие

При использовании сертификатов для аутентификации и Клиент, и Сервер должны включать свою цепочку сертификатов, хранящуюся в соответствующих учетных данных, как часть выбранного набора шифров аутентификации.

Каждое Устройство должно проверять цепочку сертификатов, представленную одноранговым Устройством. Каждая подпись сертификата должна проверяться до тех пор, пока в Ресурсе «/ oic / sec / cred» не будет найден открытый ключ с использованием «oic.sec.cred.trustca».

Устройства должны следовать алгоритму проверки пути сертификата, описанному в разделе 6 IETF RFC 5280. Кроме того:

- Для сертификатов как конечных объектов, так и сертификатов не конечных объектов Устройства должны проверять, что поля «notBefore» и «notAfter» в сертификатах соответствуют пунктам 4.1.2.5 IETF RFC 5280, 4.1.2.5.1 и 4.1.2.5.2.

- Для сертификатов, не являющихся конечными объектами, Устройства должны проверять, присутствует ли расширение базовых ограничений и что логическое значение «cA» в расширении имеет значение ИСТИНА. Если какой-либо из них неверен, цепочка сертификатов должна быть отклонена. Если присутствует поле pathLenConstraint, Устройства должны проверить, что количество сертификатов между этим сертификатом и сертификатом конечного объекта меньше или равно «pathLenConstraint». В частности, если pathLenConstraint равен нулю, этим сертификатом может быть выдан только сертификат конечного объекта. Если поле «pathLenConstraint» отсутствует, длина цепочки не ограничена.

- Для сертификатов конечных объектов Устройства должны проверять, что расширение базовых ограничений (если присутствует) имеет логическое значение «cA», равное FALSE, и не содержит последовательность ASN.1 «pathLenConstraint».

- Для сертификатов, не являющихся конечными объектами, Устройства должны проверять, присутствует ли расширение использования ключа и установлен ли бит «keyCertSign» (5).

- Для сертификатов конечных объектов Устройства должны проверять наличие расширения Key Usage и наличие битов digitalSignature (0) и keyAgreement (4).

- Для сертификатов конечных объектов Устройства должны проверять наличие расширения расширенного использования ключа (EKU) и его соответствие цели, для которой оно представлено: идентификация («1.3.6.1.4.1.44924.1.6») или роль ( «1.3.6.1.4.1.44924.1.7»). Сертификат конечного объекта, который не содержит расширения EKU или представляет идентификаторы OID как идентичности, так и роли, недействителен и должен быть отклонен. Любой сертификат, который содержит цель «anyExtendedKeyUsage» («2.5.29.37.0»), должен быть отклонен, даже если присутствуют и другие действующие EKU. Для сертификатов конечных объектов Устройства должны проверять, что расширение EKU также содержит OID для «serverAuthentication».

(«1.3.6.1.5.5.7.3.1») и «clientAuthentication» («1.3.6.1.5.5.7.3.2») для совместимости с различными реализациями TLS.

- Для сертификатов конечных объектов, которые связаны с корневым ЦС OCF, Устройства должны проверить, что они содержат по крайней мере один «PolicyIdentifierId», установленный на OID политики сертификатов OCF - («1.3.6.1.4.1.51414.0.1.2»), соответствующий версия Политики сертификации OCF, в соответствии с которой он был выпущен. Также могут быть заполнены дополнительные идентификаторы CP OID, зависящие от производителя.

Если Устройство не распознает добавочный номер, оно должно проверить «критическое» поле. Если поле имеет значение ИСТИНА, Устройство отклоняет сертификат. Если в поле установлено значение FALSE, Устройство будет обрабатывать сертификат так, как если бы расширение отсутствовало, и действовать соответствующим образом. Это относится ко всем сертификатам в цепочке.

Устройство извлекает UUID субъекта из компонента «Общее имя» свойства «Имя субъекта» сертификата конечного объекта, который имеет следующий формат: «uuid: X» ,, где X предоставляется CMS в соответствии с " deviceuuid "Свойство ресурса" / oic / sec / doxm ". Устройство рассматривает все запросы, поступающие через соединение, аутентифицированное этим сертификатом конечного объекта, как исходящие от Устройства с этим UUID субъекта. Устройство должно использовать этот UUID субъекта для сопоставления со свойством "subjectuuid" предоставленных записей ACL для выполнения проверок управления доступом.

### 10.4.2 Утверждение роли с помощью сертификатов

В этом пункте описывается утверждение роли клиентом серверу с использованием учетных данных роли сертификата.

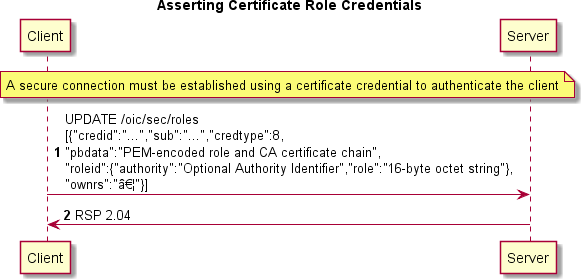
После аутентификации с помощью сертификата клиент OCF должен утверждать роли, обновляя Ресурс сервера «/ oic / sec / roles» со всеми сертификатами ролей, которыми он обладает, если только производитель устройства не предоставляет Конечному пользователю специфичный для поставщика механизм для выбора ролей. утверждать.

Учетные данные роли должны быть учетными данными сертификата и должны включать цепочку сертификатов. Сервер должен проверять каждую цепочку сертификатов, как указано в пункте 10.3. Кроме того, открытый ключ в сертификате конечного объекта, используемый для аутентификации устройства, должен быть идентичен открытому ключу во всех ролях.

(End-Entity) сертификаты. Кроме того, компонент общего имени имени субъекта для сертификатов ролей и сертификатов идентичности должен включать строку формата «uuid: X», где X соответствует свойству «deviceuuid» ресурса «oic.sec.doxm».

Кроме того, Клиенту запрещается добавлять ролевые сертификаты для других Клиентов. Сервер должен отклонить запрос клиентов на добавление сертификатов ролей, если либо (1) запрос был получен через незащищенное соединение, либо (2) запрос был получен через защищенное соединение, но открытый ключ в сертификате роли не соответствует открытый ключ в сертификате личности, который использовался для установления защищенного соединения.

Заявленные роли закодированы в расширении subjectAltName в сертификате. Поле «subjectAltName» может иметь несколько значений, что позволяет одному сертификату кодировать несколько ролей, применимых к клиенту. Сервер также должен проверить, что расширение EKU сертификата (ов) ролей содержит значение «1.3.6.1.4.1.44924.1.7» (см. Пункт 9.4.2.2), указывающее, что сертификат может использоваться для утверждения ролей. На рисунке 24 показано, как клиентское устройство утверждает роли серверу.

Рисунок 24 - Утверждение роли с учетными данными роли сертификата.

Дополнительные комментарии к рисунку 24

1) Ответ должен содержать «204 No Content» для обозначения успеха или 4xx для обозначения ошибки. Если сервер не поддерживает учетные данные сертификата, он должен вернуть «501 Not Implemented».

2) Роли, заявленные клиентом, могут сохраняться в течение периода, выбранного сервером. Срок действия не должен превышать срок действия ролевого сертификата.

3) Серверы должны выбирать ненулевую продолжительность, чтобы избежать затрат на частое повторное утверждение роли клиентом. Рекомендуется, чтобы серверы использовали срок действия сертификата в качестве продолжительности, эффективно позволяя CMS определять продолжительность.

4) Формат данных, отправляемых в вызове create, должен быть списком учетных данных («oic.sec.cred», см. Таблицу 19). Они должны иметь "credtype" 8 (с указанием сертификатов), а поле "PrivateData" не должно присутствовать. Для полей, которые дублируются в объекте «oic.sec.cred» и сертификате, значение в сертификате должно использоваться для проверки. Например, если в учетных данных установлено поле «Период», сервер должен рассматривать срок действия сертификата как авторитетный. Аналогично для данных roleid (авторитет, роль).

5) Сертификаты должны быть закодированы, как показано на рисунке 23 (цепочка сертификатов в кодировке PEM).

6) Клиенты могут ПОЛУЧИТЬ ресурс «/ oic / sec / roles» для определения ранее заявленных ролей. Должен быть возвращен массив объектов учетных данных. Если нет действительных сертификатов, соответствующих текущему подключенному и аутентифицированному клиенту, то должен быть возвращен пустой массив (то есть []).

### 10.4.3 Корни OCF PKI

Этот пункт намеренно оставлен пустым.

### 10.4.4 Хранилище доверенных сертификатов PKI

Каждое Устройство, использующее сертификат, связанный с привязкой доверия корневого ЦС OCF, ДОЛЖНО надежно хранить сертификаты корневого ЦС OCF в Ресурсе «oic / sec / cred» и ДОЛЖНО физически хранить этот Ресурс в защищенной области памяти, где сертификаты не могут быть изменены.

### 10.4.5 Проверка пути и обработка расширений

См. Пункт 10.3.

# 11 Целостность и конфиденциальность сообщений

## 11.1 Преамбула

Защищенная связь между Клиентами и Серверами защищена от перехвата, подделки или повторного воспроизведения сообщений с использованием механизмов безопасности, которые обеспечивают конфиденциальность и целостность сообщений.

## 11.2 Защита сеанса с помощью DTLS

### 11.2.1 Общие сведения о защите DTLS

Устройства должны поддерживать DTLS для защищенной связи, как определено в IETF RFC 6347. Устройства использование TCP должно поддерживать TLS v1.2 для защищенной связи, как определено в IETF RFC 5246. См. 11.3 для списка обязательных и дополнительных комплектов шифров для передачи сообщений.

Устройства OCF ДОЛЖНЫ поддерживать (D) TLS версии 1.2 или выше и НЕ ДОЛЖНЫ поддерживать версии 1.1 или ниже.

Семантика многоадресного сеанса еще не определена в этой версии документа по безопасности.

### 11.2.2 Семантика одноадресного сеанса

Для одноадресных сообщений между Клиентом и Сервером оба Устройства должны аутентифицировать друг друга.

См. Раздел 10 для получения подробной информации об аутентификации устройства.

Защищенные одноадресные сообщения между клиентом и сервером должны использовать набор шифров из 11.3.

Отправляющее Устройство должно шифровать и аутентифицировать сообщения, как определено выбранным шифром.

Suite и принимающее Устройство должны проверять и расшифровывать сообщения перед их обработкой.

## 11.3 Наборы шифров

### 11.3.1 Общие наборы шифров

Наборы шифров, разрешенные для использования, могут различаться в зависимости от контекста. В этом разделе перечислены наборы шифров, разрешенные во время передачи права собственности и нормальной работы. Следующие RFC предоставляют дополнительную информацию о наборах шифров, используемых в OCF.

IETF RFC 4279: определяет использование предварительно общих ключей (PSK) в (D) TLS

IETF RFC 4492: определяет использование криптографии эллиптических кривых в (D) TLS

IETF RFC 5489: определяет использование наборов шифров, которые используют эллиптическую кривую Диффи-Хеллмана (ECDHE) и PSK.

IETF RFC 6655 и IETF RFC 7251: Определяет комплекты шифров для режима AES-CCM с ECDHE.

### 11.3.2 Наборы шифров для передачи права собственности на устройство

#### 11.3.2.1 Наборы шифров для метода Just Works

Just Works OTM может использовать следующие (D) наборы шифров TLS.

TLS\_ECDH\_ANON\_WITH\_AES\_128\_CBC\_SHA256

Все Устройства, поддерживающие Just Works OTM, должны реализовывать:

TLS\_ECDH\_ANON\_WITH\_AES\_128\_CBC\_SHA256 (со значением 0xFF00)

#### 11.3.2.2 Наборы шифров для метода случайного ПИН-кода

OTM на основе случайного ПИН-кода может использовать следующие (D) наборы шифров TLS.

TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256

Все устройства, поддерживающие OTM на основе случайных выводов, должны реализовывать:

TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256

#### 11.3.2.3 Наборы шифров для метода сертификата

OTM на основе сертификата производителя может использовать следующие (D) наборы шифров TLS.

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM

Используя следующую кривую:

secp256r1 (см. IETF RFC 4492)

Все устройства, поддерживающие OTM на основе сертификатов производителя, должны реализовывать:

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8

Устройства, поддерживающие OTM на основе сертификатов производителя, должны реализовывать:

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM

11.3.3 Наборы шифров для симметричных ключей

Следующие наборы шифров определены для (D) TLS-связи с использованием PSK:

TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,

TLS\_PSK\_WITH\_AES\_128\_CCM\_8, (\* 8 тегов аутентификации OCTET \*)

TLS\_PSK\_WITH\_AES\_256\_CCM\_8,

TLS\_PSK\_WITH\_AES\_128\_CCM, (\* 16 тегов аутентификации OCTET \*)

TLS\_PSK\_WITH\_AES\_256\_CCM,

Все наборы шифров на основе CCM также используют HMAC-SHA-256 для аутентификации.

Все Устройства должны реализовывать следующее:

TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,

В устройствах должно быть реализовано следующее:

TLS\_ECDHE\_PSK\_WITH\_AES\_128\_CBC\_SHA256,

TLS\_PSK\_WITH\_AES\_128\_CCM\_8,

TLS\_PSK\_WITH\_AES\_256\_CCM\_8,

TLS\_PSK\_WITH\_AES\_128\_CCM,

TLS\_PSK\_WITH\_AES\_256\_CCM

#### 11.3.4 Наборы шифров для асимметричных учетных данных

Следующие наборы шифров определены для (D) TLS-связи с асимметричным

сертификаты:

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM

Используя следующую кривую:

secp256r1 (см. IETF RFC 4492)

Все устройства, поддерживающие асимметричные учетные данные, должны реализовывать:

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM\_8

Все устройства, поддерживающие асимметричные учетные данные, должны реализовывать:

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM\_8,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CCM,

TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CCM

# 12 Контроль доступа

## 12.1 Создание и управление ACL

Этот пункт намеренно оставлен пустым.

## 12.2 Оценка и применение ACL

### 12.2.1 Оценка и применение ACL Общие

Сервер обеспечивает контроль доступа к Ресурсам приложения перед тем, как предоставить их запрашивающей стороне. Уровень безопасности на сервере аутентифицирует инициатора запроса, когда доступ получен через защищенный порт. Аутентифицированные запрашивающие, известные как "субъект", могут использоваться для сопоставления записей ACL, которые определяют личность, роль запрашивающей стороны или могут сопоставлять аутентифицированные запрашивающие, используя подстановочный знак субъекта.

Если запрос поступает через незащищенный порт, единственными разрешенными политиками ACL являются те, которые используют соответствие подстановочного знака субъекта анонимным запросчикам.

В доступе отказано, если запрошенный ресурс не соответствует записи ACL.

ПРИМЕЧАНИЕ. Существуют задокументированные исключения, относящиеся к подключению устройства, когда доступ к виртуальным ресурсам безопасности может быть предоставлен до предоставления ресурсов ACL.

ACL второго поколения (то есть «/ oic / sec / acl2») содержит массив записей управления доступом (ACE2), которые используют алгоритм сопоставления ресурсов, который использует массив ссылок на ресурсы для сопоставления ресурсов, к которым применяется политика доступа ACE2. Сопоставление состоит из сравнения значений свойства «ресурсы» ACE2 (см. Раздел 13) с запрошенным ресурсом. Ресурсы подбираются двумя способами:

1) ссылка на хост ("href")

2) Подстановочный знак ресурса («wc»).

### 12.2.2 Сопоставление ссылок на хост

При наличии в элементе сопоставления ACE2 свойство ссылки на хост (href) должно использоваться для сопоставления ресурсов.

- Свойство href должно использоваться для поиска точного соответствия имени ресурса, если оно присутствует.

### 12.2.3 Сопоставление подстановочных знаков ресурса

Если присутствует, выражение с подстановочными знаками ("wc") должно использоваться для сопоставления нескольких ресурсов с использованием

wildcard Свойство, содержащееся в структуре «oic.sec.ace2.resource-ref».

Выражение с подстановочным знаком может использоваться для сопоставления нескольких ресурсов с помощью свойства подстановочного знака.

содержится в структуре «oic.sec.ace2.resource-ref». Строки сопоставления с подстановочными знаками определены

в таблице 14.

Таблица 14 - Описание строк соответствия подстановочных знаков ACE2

|  |  |
| --- | --- |
| **String** | **Description** |
| "+" | Shall match all Discoverable Non-Configuration Resources which expose at least one Secure OCF Endpoint. |
| "-" | Shall match all Discoverable Non-Configuration  Resources which expose at least one Unsecure OCF Endpoint. |
| "\*" | Shall match all Non-Configuration Resources. |

NOTE Discoverable Resources appear in the "/oic/res" Resource, while non-discoverable Resources may appear in other collection Resources but do not appear in the /res collection.

### 12.2.4 Multiple Criteria Matching

|  |
| --- |
| If the ACE2 "resources" Property contains multiple entries, then a logical OR shall be applied for |
| each array element. For example, if a first array element of the "resources" Property contains |
| "href"="/a/light" and the second array element of the "resources" Property contains "href"="/a/led", |
| then Resources that match either of the two "href" criteria shall be included in the set of matched |
| Resources. |
| Example 1 JSON for Resource matching |
| { |
| //Matches Resources named "/x/door1" or "/x/door2" |
| "resources":[ |
| { |
| "href":"/x/door1" |
| }, |
| { |
| "href":"/x/door2" |
| }, |
| ] |
| } |
| Example 2 JSON for Resource matching |
| { |
| // Matches all Resources |
| "resources":[ |
| { |
| "wc":"\*" |
| } |
| ] |
| } |
| 12.2.5 Subject Matching using Wildcards |
| When the ACE subject is specified as the wildcard string "\*" any requestor is matched. The OCF |
| server may authenticate the OCF client, but is not required to. |
| Examples: JSON for subject wildcard matching |
| //matches all subjects that have authenticated and confidentiality protections in place. |
| "subject" : { |
| "conntype" : "auth-crypt" |
| } |
| //matches all subjects that have NOT authenticated and have NO confidentiality protections in place. |
| "subject" : { |
| "conntype" : "anon-clear" |
| } |
| **12.2.6 Subject Matching using Roles** |
| When the ACE subject is specified as a role, a requestor shall be matched if either: |
| 1) The requestor authenticated with a symmetric key credential, and the role is present in the |
| "roleid" Property of the credential’s entry in the "credential" Resource, or |
| 2) The requestor authenticated with a certificate, and a valid role certificate is present in the roles |
| Resource with the requestor’s certificate’s public key at the time of evaluation. Validating role |
| certificates is defined in 10.3.1. |
| **12.2.7 ACL Evaluation** |
| **12.2.7.1 ACE2 matching algorithm** |
| The OCF Server shall apply an ACE2 matching algorithm that matches in the following sequence: |
| 1) The local "/oic/sec/acl2" Resource contributes its ACE2 entries for matching. |
| 2) Access shall be granted when all these criteria are met: |
| a) The requestor is matched by the ACE2 "subject" Property. |
| b) The requested Resource is matched by the ACE2 "resources" Property and the requested |
| Resource shall exist on the local Server. |
| c) The "period" Property constraint shall be satisfied. |
| d) The "permission" Property constraint shall be applied. |
| If multiple ACE2 entries match the Resource request, the union of permissions, for all matching |
| ACEs, defines the effective permission granted. E.g. If Perm1=CR---; Perm2=--UDN; Then UNION |
| (Perm1, Perm2)=CRUDN. |
| The Server shall enforce access based on the effective permissions granted. |
| Batch requests to Resource containing Links require additional considerations when accessing the |
| linked Resources. ACL considerations for batch request to the Atomic Measurement Resource |
| Type are provided in clause [12.2.7.2.](#_bookmark204) ACL considerations for batch request to the Collection |
| Resource Type are provided in clause [12.2.7.3.](#_bookmark205) |
| Clause [12.2.7.4](#_bookmark206) provides ACL considerations when a new Resource is created on a Server in |
| response to a CREATE request. |
| **12.2.7.2 ACLconsiderations for batch request to the Atomic Measurement Resource Type** |
| The present clause shall apply to any Resource Type based on the Atomic Measurement Resource |
| Type. |
| If an OCF Server receives a batch OCF Interface request to an Atomic Measurement Resource and |
| there is an ACE matching the Atomic Measurement Resource which permits the request, then the |
| corresponding requests to the linked Resources of the Atomic Measurement Resource shall be |
| permitted by the OCF Server. That is, the request to each linked Resource is permitted regardless |
| of whether there is an ACE configured on the OCF Server which would permit a corresponding |
| request from the OCF Client (which sent the batch OCF Interface request to the Atomic |
| Measurement Resource) addressing the linked Resource. |
| NOTE As specified in [ISO/IEC 30118-1,](#_bookmark3) the linked Resources of an Atomic Measurement Resource are hosted on the |
| same Device as the Atomic Measurement Resource. |
| **12.2.7.3 ACL considerations for a batch OCF Interface request to a Collection** |
| This cluase addresses the additional authorization processes which take place when a Server |
| receives a batch OCF Interface request from a Client to a Collection hosted on that Server, |
| assuming there is an ACE matching the Collection which permits the original Client request. For |
| the purposes of this cluase, the Server hosting this Collection is called the "Collection host". The |
| additional authorization process is dependent on whether the linked Resource is hosted on the |
| Collection host or the linked Resource is hosted on another Server: |
| – For each generated request to a linked Resource hosted on the Collection host, the Collection |
| host shall apply the ACE2 matching algorithm in clause [12.2.7.1](#_bookmark203) to determine whether the linked |
| Resource is permitted to process the generated request, with the following clarifications: |
| – The requestor in clause [12.2.7.1](#_bookmark203) shall be the Client which sent the original Client request. | |
| – The requested Resource in clause [12.2.7.1](#_bookmark203) shall be the linked Resource, which shall be | |
| matched using at least one of: | |
| – a Resource Wildcard matching the linked Resource, or | |
| – an exact match of the local path of the linked Resource with a "href" Property in the | |
| "resources" array in the ACE2. | |
| – an exact match of the full URI of the linked Resource with a "href" Property in the | |
| "resources" array in the ACE2. | |
| NOTE The full URI of a linked Resource is obtained by concatenating the "anchor" Property of the Link, if present, and | |
| the "href" Property of the Link. The local path can then be determined form the full URI. | |
| If the linked Resource is not permitted to process the generated request, then the Collection host | |
| shall treat such cases as a linked Resource which cannot process the request when composing the | |
| aggregated response to the original Client Request, as specified for the batch OCF Interface in the | |
| [ISO/IEC 30118-1.](#_bookmark3) | |
| **12.2.7.4 ACL Considerations on creation of a new Resource** | |
| When a new Resource is created on a Server in response to a CREATE request, there might be | |
| no ACEs permitting access to the newly created Resource. The present clause describes how the | |
| Server autonomously modifies the "/oic/sec/acl2" Resource to provide some initial authorizations | |
| for accessing the newly created Resource. The purpose of this autonomous modification is to avoid | |
| relying on the AMS update the "/oic/sec/acl2" Resource after every new Resource is created. | |
| Subsequent to a Server creating a Collection inside another Collection in response to a CREATE | |
| request from a Client, and prior to sending a response to the Client: | |
| – If there is an ACE with "subject" containing the UUID of the Client, and "permissions" exactly | |
| matching the CREATE, RETRIEVE, UPDATE and DELETE operations, then the Server shall | |
| autonomously add an "href" entry to "resources" with the URI of the newly created Collection. | |
| – Otherwise, the Server shall autonomously add an ACE with "subject" containing the UUID | |
| of the Client, "resources" containing an "href" entry with the URI of the newly created | |
| Collection, and "permissions" exactly matching the CREATE, RETRIEVE, UPDATE and | |
| DELETE operations. | |
| Subsequent to a Server creating a non-Collection Resource inside another Collection in response | |
| to a CREATE request from a Client, and prior to sending a response to the Client: | |
| – If there is an ACE with "subject" containing the UUID of the Client, and "permissions" exactly | |
| matching the RETRIEVE, UPDATE and DELETE operations, then the Server shall | |
| autonomously add an "href" entry to "resources" with the URI of the newly created Resource. | |
| – Otherwise, the Server shall autonomously add an ACE with "subject" containing the UUID | |
| of the Client, "resources" containing an "href" entry with the URI of the newly created, and | |
| "permissions" exactly matching the RETRIEVE, UPDATE and DELETE operations. | |
|  | |

# 13. Security Resources

##### Security Resources General

OCF Security Resources are shown in [Figure 25.](#_bookmark209)

"/oic/sec/cred" Resource and Properties are shown in [Figure 26.](#_bookmark210) "/oic/sec/acl2" Resource and Properties are shown in [Figure 27.](#_bookmark211)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **"/oic/sec/doxm" Resource**  --------------------  oxm oxmsel sct  owned  deviceuuid  devowneruuid rowneruuid | **"/oic/sec/cred" Resource**  --------------------  creds  rowneruuid | **"/oic/sec/acl2" Resource**  --------------------  aclist2  rowneruuid | **"/oic/sec/pstat" Resource**  --------------------  dos isop cm tm  om sm  rowneruuid | **"/oic/sec/roles" Resource**  --------------------  roles |
|  |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **"/oic/sec/csr" Resource**  --------------------  csr  encoding | **"/oic/sec/sp" Resource**  --------------------  currentprofile supportedprofiles | **"/oic/sec/ael" Resource**  --------------------  events  usedspace maxspace unit  categoryfilter priorityfilter | **"/oic/sec/sdi" Resource**  --------------------  uuid name priv |
|  | |
|  |

##### Figure 25 – OCF Security Resources

creds

rowneruuid

**"/oic/sec/cred" Resource**

|  |  |
| --- | --- |
| **"creds"** |  |
| **Property** |
| credid |
| subjectuuid |
| roleid |
|  |
| credtype |
| credusage |
| publicdata |
| privatedata |
| optionaldata |
| period |
| cms |
|  |  |

##### Figure 26 – "/oic/sec/cred" Resource and Properties

encoding data

revstat

**"optdatatype"**

**Property**

encoding data

handle

**"privdatatype" Property**

href rt

if wc

**"resource" Property**

aclist2

rowneruuid

**"/oic/sec/acl2" Resource**

subject

resources permission validity

aceid

**"aclist2" Property**

didtype conntype roletype

**"subject" Property**

##### Figure 27 – "/oic/sec/acl2" Resource and Properties

aeid

devicetype di

category priority

timestamp message auxiliaryinfo

**"events" Property**

events

usedspace maxspace unit

categoryfilter priorityfilter

**"/oic/sec/ael" Resource**

|  |  |
| --- | --- |
| 2791 | **Figure 28 – "/oic/sec/ael" Resource and Properties** |
| 2792 | **13.2 Device Owner Transfer Resource** |
| 2793 | **13.2.1 Device Owner Transfer Resource General** |
| 2794 | The "/oic/sec/doxm" Resource contains the set of supported Device OTMs. |
| 2795 | Resource discovery processing respects the CRUDN constraints supplied as part of the security |
| 2796 | Resource definitions contained in this document. |
| 2797 | "/oic/sec/doxm" Resource is defined in [Table 15.](#_bookmark215) |
| 2798 | **Table 15 – Definition of the "/oic/sec/doxm" Resource** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/doxm | Device OTMs | oic.r.doxm | oic.if.baselin e, oic.if.rw | Resource for supporting Device owner transfer | Configuration |

[Table 16](#_bookmark216) defines the Properties of the "/oic/sec/doxm" Resource.

##### Table 16 – Properties of the "/oic/sec/doxm" Resource

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandat ory** | **Device State** | **Access Mode** | **Description** |
| OTM | oxms | oic.sec.doxmt ype | array | Yes |  | R | Value identifying the owner-transfer- method and the organization that  defined the method. |
| OTM  Selection | oxmsel | oic.sec.doxmt ype | UINT16 | Yes | RESET | R | Server shall set to (4) "oic.sec.oxm.self" |
| RFOTM  (no open DOC) | RW | DOTS shall set to its selected DOTS and both parties execute the DOTS. After secure owner transfer session is established DOTS shall update the  oxmsel again making it permanent. If the |
|  |  |  |  |  |  |  | DOTS fails the Server shall transition device state to RESET. |
| RFOTM  (open DOC) | R | n/a |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | R | n/a |
| Supported Credential Types | sct | oic.sec.credty pe | bitmask | Yes |  | R | Identifies the types of credentials the Device supports. The Server sets this value at framework initialization after determining security capabilities.  The Device always supports symmetric pair-wise key and asymmetric signing key with certificate (bit positions 0x1  and 0x8 respectively). Other credential types are optional as per clause [9.3](#_bookmark138) |
| Device Ownership Status | owned | Boolean | T|F | Yes | RESET | R | Server shall set to FALSE. |
| RFOTM  (no open DOC) | R | FALSE |
| RFOTM  (open DOC) | RW | DOTS (Device communicating over DOC) shall set to TRUE after secure owner transfer session is established. |
| RFPRO | R | TRUE |
| RFNOP | R | TRUE |
| SRESET | R | TRUE |
| Device UUID | deviceuuid | String | oic.sec.didt ype | Yes | RESET | R | No stipulation. |
| RFOTM  (no open DOC) | R | n/a |
| RFOTM  (open DOC) | RW | DOTS (Device communicating over  DOC) updates to a value it has selected after secure owner transfer session is  established. |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | R | n/a |
| Device Owner Id | devowneruu id | String | uuid | Yes | RESET | R | Server shall set to the nil uuid value (e.g. "00000000-0000-0000-0000-  000000000000" ) |
| RFOTM  (no open DOC) | R | n/a |
| RFOTM  (open DOC) | RW | DOTS (Device communicating over  DOC) shall set value after secure owner transfer session is established. |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | R | n/a |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Resource Owner Id | rowneruuid | String | uuid | Yes | RESET | R | Server shall set to the nil uuid value (e.g. "00000000-0000-0000-0000-  000000000000" ) |
| RFOTM  (no open DOC) | R | n/a |
| RFOTM  (open DOC) | RW | The DOTS (Device communicating over DOC) shall configure the rowneruuid Property when a successful owner  transfer session is established. |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | RW | The DOTS (referenced via  devowneruuid Property) should verify and if needed, update the Resource owner Property when a mutually  authenticated secure session is  established. If the rowneruuid does not refer to a valid DOTS device identifier the Server shall transition to RESET. |

[Table 17](#_bookmark217) defines the Properties of the "oic.sec.didtype".

##### Table 17 – Properties of the "oic.sec.didtype" type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mand atory** | **Device State** | **Access Mode** | **Description** |
| Device UUID | uuid | String | uuid | Yes | RW | - | A uuid value |

|  |
| --- |
| The "oxms" Property contains a list of OTM where the entries appear in the order of preference. |
| This Property contains the higher priority methods appearing before the lower priority methods. |
| The DOTS queries this list at the time of onboarding and selects the most appropriate method. |
| OTMs consist of two parts, a URI identifying the vendor or organization and the specific method. |
| <DoxmType> ::= <NSS> |
| <NSS> ::= <Identifier> | {{<NID>"."} <NameSpaceQualifier> "."} <Method> |
| <NID> :: = <Vendor-or-Organization> |
| <Identifier> ::= INTEGER |
| <NameSpaceQualifier> ::= String |
| <Method> ::= String |
| <Vendor-Organization> ::= String |
| When an OTM successfully completes, the "owned" Property is set to "1" (TRUE). Consequently, |
| subsequent attempts to take ownership of the Device will fail. |
| There are four device identifiers: |
| 1) "deviceuuid" Property of "/oic/sec/doxm" Resource - random DOTS-provisioned value unique |
| for a given security domain, used as a device identity for access control, mapped internally to |
| a device-owned credential. |
| 2) "di" Property of "/oic/d" Resource - mirroring the value of "deviceuuid" Property of |
| "/oic/sec/doxm" Resource. |
| 3) "piid" Property of "/oic/d" Resource - defined in [ISO/IEC 30118-1.](#_bookmark3) |
| 4) "pi" Property of "/oic/p" Resource - defined in [ISO/IEC 30118-1.](#_bookmark3) |
| The "/oic/sec/doxm" Resource supports CoAP multicast requests in certain cases. For details see |
| clause [7.3.1](#_bookmark96) |

##### 13.2.2 OCF defined OTMs

[Table 18](#_bookmark219) defines the Properties of the "oic.sec.doxmtype".

##### Table 18 – Properties of the "oic.sec.doxmtype" type

|  |  |  |  |
| --- | --- | --- | --- |
| **Value Type Name** | **Value Type URN (optional)** | **Enumeration Value**  **(mandatory)** | **Description** |
| OCFJustWorks | oic.sec.doxm.jw | 0 | The just-works method relies on anonymous Diffie-  Hellman key agreement protocol to allow a DOTS to  assert ownership of the new Device. The first DOTS to make the assertion is accepted as the Device owner. The just-works method results in a shared secret that is used to authenticate the Device to the DOTS and likewise authenticates the DOTS to the Device. The Device permits the DOTS to take ownership of the  Device, after which a second attempt to take ownership by a different DOTS will faila. |
| OCFSharedPin | oic.sec.doxm.rdp | 1 | The new Device randomly generates a PIN that is communicated via an Out Of Band Communication Channel to a DOTS. An in-band Diffie-Hellman key agreement protocol establishes that both endpoints  possess the PIN. Possession of the PIN by the DOTS signals the new Device that device ownership can be asserted. |
| OCFMfgCert | oic.sec. doxm.mfgcert | 2 | The new Device is presumed to have been  manufactured with an embedded asymmetric private key that is used to sign a Diffie-Hellman exchange at Device onboarding. The manufacturer certificate  should contain Platform hardening information and other security assurances assertions. |
| OCF Reserved | <Reserved> | 3 | Reserved |
| OCFSelf | oic.sec.oxm.self | 4 | The manufacturer shall set the "/doxm.oxmsel" value to (4). The Server shall reset this value to (4) upon  entering RESET. |
| OCF Reserved | <Reserved> | 5~0xFEFF | Reserved for OCF use |
| Vendor-defined Value Type Name | <Reserved> | 0xFF00~0xFFFF | Reserved for vendor-specific OTM use |
| a The just-works method is subject to a man-in-the-middle attacker. Precautions should be taken to provide physical security when this method is used. | | | |

|  |
| --- |
| **13.3 Credential Resource** |
| **13.3.1 Credential Resource General** |
| The "/oic/sec/cred" Resource maintains credentials used to authenticate the Server to Clients and |
| support services as well as credentials used to verify Clients and support services. |
| Multiple credential types are anticipated by the OCF framework, including pair-wise pre-shared |
| keys, asymmetric keys, certificates and others. The credential Resource uses a Subject UUID to |
| distinguish the Clients and support services it recognizes by verifying an authentication challenge. |
| In order to provide an interface which allows management of the "creds" Array Property, the |
| RETRIEVE, UPDATE and DELETE operations on the "/oic/sec/cred" Resource shall behave as |
| follows: |
| 1) A RETRIEVE shall return the full Resource representation, except that any write-only Properties |
| shall be omitted (e.g. private key data). |
| 2) An UPDATE shall replace or add to the Properties included in the representation sent with the |
| UPDATE request, as follows: |

|  |  |
| --- | --- |
| 2843 | a) If an UPDATE representation includes the "creds" array Property, then: |
| 2844 | i) Supplied "creds" with a "credid" that matches an existing "credid" shall replace |
| 2845 | completely the corresponding "cred" in the existing "creds" array. |
| 2846 | ii) Supplied "creds" without a "credid" shall be appended to the existing "creds" array, and |
| 2847 | a unique (to the "cred" Resource) "credid" shall be created and assigned to the new |
| 2848 | "cred" by the Server. The "credid" of a deleted "cred" should not be reused, to improve |
| 2849 | the determinism of the interface and reduce opportunity for race conditions. |
| 2850 | iii) Supplied "creds" with a "credid" that does not match an existing "credid" shall be |
| 2851 | appended to the existing "creds" array, using the supplied "credid". |
| 2852 | iv) The rows in [Table 20](#_bookmark223) corresponding to the "creds" array Property dictate the Device |
| 2853 | States in which an UPDATE of the "creds" array Property is always rejected. If OCF |
| 2854 | Device is in a Device State where the Access Mode in this row contains "R", then the |
| 2855 | OCF Device shall reject all UPDATEs of the "creds" array Property. |
| 2856 | 3) A DELETE without query parameters shall set the "creds" array to the empty array, but shall |
| 2857 | not remove the "/oic/sec/cred" Resource. |
| 2858 | 4) A DELETE with one or more "credid" query parameters shall remove the "cred"(s) with the |
| 2859 | corresponding "credid"(s) from the "creds" array. |
| 2860 | 5) The rows in [Table 20](#_bookmark223) corresponding to the "creds" array Property dictate the Device States in |
| 2861 | which a DELETE is always rejected. If OCF Device is in a Device State where the Access Mode |
| 2862 | in this row contains "R", then the OCF Device shall reject all DELETEs. |
| 2863 | NOTE The "/oic/sec/cred" Resource's use of the DELETE operation is not in accordance with the OCF Interfaces defined |
| 2864 | in [ISO/IEC 30118-1.](#_bookmark3) |
| 2865 | "/oic/sec/cred" Resource is defined in [Table 19.](#_bookmark222) |
| 2866 | **Table 19 – Definition of the "/oic /sec/cred" Resource** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/cred | Credentials | oic.r.cred | oic.if.baseline, oic.if.rw | Resource containing credentials for Device  authentication, verification and data protection | Security |

[Table 20](#_bookmark223) defines the Properties of the "/oic/sec/cred" Resource.

##### Table 20 – Properties of the "/oic/sec/cred" Resource

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandat ory** | **Device State** | **Access Mode** | **Description** |
| Credentials | creds | oic.sec.cre d | array | Yes | RESET | R | Server shall set to manufacturer defaults. |
| RFOTM | RW | Set by DOTS after successful OTM |
| RFPRO | RW | Set by the CMS (referenced via the rowneruuid Property of "/oic/sec/cred"  Resource) after successful  authentication. Access to NCRs is prohibited. |
| RFNOP | R | Access to NCRs is permitted after a matching ACE is found. |
| SRESET | RW | The DOTS (referenced via devowneruuid Property of  "/oic/sec/doxm" Resource or the rowneruuid Property of  "/oic/sec/doxm" Resource) should evaluate the integrity of and may  update creds entries when a secure session is established and the Server and DOTS are authenticated. |
| Resource Owner ID | rowneruuid | String | uuid | Yes | RESET | R | Server shall set to the nil uuid value (e.g. "00000000-0000-0000-0000-  000000000000" ) |
| RFOTM | RW | The DOTS shall configure the  rowneruuid Property of "/oic/sec/cred"  Resource when a successful owner transfer session is established. |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | RW | The DOTS (referenced via devowneruuid Property of  "/oic/sec/doxm" Resource or the rowneruuid Property of  "/oic/sec/doxm" Resource) should verify and if needed, update the Resource owner Property when a  mutually authenticated secure session is established. If the "rowneruuid"  Property does not refer to a valid DOTS the Server shall transition to RESET. |

All secure Device accesses shall have a "/oic/sec/cred" Resource that protects the end-to-end interaction.

The "/oic/sec/cred" Resource shall be updateable by the service named in its rowneruuid Property.

ACLs naming "/oic/sec/cred" Resource should further restrict access beyond CRUDN access modes.

[Table 21](#_bookmark224) defines the Properties of "oic.sec.creds".

2875

##### Table 21 – Properties of the "oic.sec.creds" Property

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandat ory** | **Access Mode** | **Device State** | **Description** |
| Credential ID | credid | UINT16 | 0 – 64K-  1 | Yes | RW |  | Short credential ID for local references from other Resource |
| Subject UUID | subjectuuid | String | uuid | Yes | RW |  | A uuid that identifies the subject to which this credential applies or "\*" if any identity is acceptable |
| Role ID | roleid | oic.sec. roletyp e | - | No | RW |  | Identifies the role(s) the subject is authorized to assert. |
| Credential Type | credtype | oic.sec. credtyp e | bitmask | Yes | RW |  | Represents this credential’s type. 0 – Used for testing  1 – Symmetric pair-wise key 2 – Symmetric group key  4 – Asymmetric signing key  8 – Asymmetric signing key with certificate 16 – PIN or password  32 – Asymmetric encryption key  64 – Directly Provisioned OSCORE Security Context  128 – Simple Secure Multicast Client Context  256 – Simple Secure Multicast Server Context |
| Credential Usage | credusage | oic.sec. credus agetyp e | String | No | RW |  | Used to resolve undecidability of the credential. Provides indication for  how/where the cred is used  "oic.sec.cred.trustca": certificate trust anchor  "oic.sec.cred.cert": identity certificate "oic.sec.cred.rolecert": role certificate  "oic.sec.cred.mfgtrustca": manufacturer certificate trust anchor  "oic.sec.cred.mfgcert": manufacturer certificate |
| Public Data | publicdata | oic.sec. pubdat atype | - | No | RW |  | Credential Type dependent. Public credential information  1:2: ticket, public SKDC values 4, 32: Public key value  8: A chain of one or more certificate |
| Private Data | privatedata | oic.sec. privdat atype | - | No | - | RESET | Server shall set to manufacturer default |
| RW | RFOTM | Set by DOTS after successful OTM |
| W | RFPRO | Set by authenticated DOTS or CMS |
| - | RFNOP | Not writable during normal operation. |
| W | SRESET | DOTS may modify to enable transition to RFPRO. |
| Optional Data | optionaldata | oic.sec. optdata type | - | No | RW |  | Credential Type dependent. Credential revocation status information  1, 2, 4, 32, 64: revocation status information  8: Revocation information |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Period | period | String | - | No | RW |  | Period as defined by [IETF RFC 5545.](#_bookmark18) The credential should not be used if the current time is outside the Period window. |
| Credential Refresh Method | crms | oic.sec. crmtyp e | array | No | RW |  | Credentials with a Period Property are refreshed using the credential refresh method (crm) according to the type  definitions for "oic.sec.crm". |
| OSCORE  Configuration | oscore | oic.sec. oscoret ype |  | No | RW |  | Contains parameters for use with  credentials intended for use with OSCORE. See type definition for "oic.sec.oscoretype" |

[Table 22](#_bookmark225) defines the Properties of "oic.sec.credusagetype".

##### Table 22: Properties of the "oic.sec.credusagetype" Property

|  |  |
| --- | --- |
| **Value Type Name** | **Value Type URN (mandatory)** |
| Trust Anchor | oic.sec.cred.trustca |
| Certificate | oic.sec.cred.cert |
| Role Certificate | oic.sec.cred.rolecert |
| Manufacturer Trust CA | oic.sec.cred.mfgtrustca |
| Manufacturer CA | oic.sec.cred.mfgcert |

[Table 23](#_bookmark226) defines the Properties of "oic.sec.pubdatatype".

##### Table 23 – Properties of the "oic.sec.pubdatatype" Property

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| Encoding format | encoding | String | N/A | RW | No | A string specifying the encoding format of the data contained in the pubdata  "oic.sec.encoding.pem" – Encoding for PEM- encoded certificate or chain |
| Data | data | String | N/A | RW | No | The encoded value |

[Table 24](#_bookmark227) defines the Properties of "oic.sec.privdatatype".

##### Table 24 – Properties of the "oic.sec.privdatatype" Property

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| Encoding format | encoding | String | N/A | RW | Yes | A string specifying the encoding format of the data contained in the privdata  "oic.sec.encoding.pem" – Encoding for PEM- encoded private key  "oic.sec.encoding.base64" – Encoding of Base64 encoded PSK  "oic.sec.encoding.handle" – Data is contained in a storage sub-system referenced using a handle  "oic.sec.encoding.raw" – Raw hex encoded data |
| Data | data | String | N/A | W | No | The encoded value  This value shall not be RETRIEVE-able. |
| Handle | handle | UINT16 | N/A | RW | No | Handle to a key storage Resource |

[Table 25](#_bookmark228) defines the Properties of "oic.sec.optdatatype".

##### Table 25 – Properties of the "oic.sec.optdatatype" Property

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| Revocation status | revstat | Boolean | T | F | RW | Yes | Revocation status flag True – revoked  False – not revoked |
| Encoding format | encoding | String | N/A | RW | No | A string specifying the encoding format of the data contained in the optdata  "oic.sec.encoding.pem" – Encoding for PEM- encoded certificate or chain |
| Data | data | String | N/A | RW | No | The encoded structure |

[Table 26](#_bookmark229) defines the Properties of "oic.sec.roletype".

##### Table 26 – Definition of the "oic.sec.roletype" type.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| Authority | authority | String | N/A | R | No | A name for the authority that defined the role. If not present, the credential issuer defined the role. If present, must be  expressible as an ASN.1 PrintableString. |
| Role | role | String | N/A - | R | Yes | An identifier for the role. Must be expressible as an ASN.1 PrintableString. |

[Table 27](#_bookmark230) defines the Properties of "oic.sec.oscoretype".

##### Table 27 – Definition of the "oic.sec.oscoretype" type.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| OSCORE  Sender ID | senderid | String | Hexadeci mal  encoding | RW | No | OSCORE Sender ID for this OSCORE Security Context. |
| OSCORE  Recipient ID | recipientid | String |  | RW | No | OSCORE Recipient ID for this OSCORE Security Context. |
| OSCORE  Sender Sequence Number 1 | ssn | Integer |  | R | No | OSCORE Sender Sequence Number being stored in non volatile memory to handle the loss of mutable security context parameters. See clause [16.2.4.](#_bookmark335) |
| OSCORE  Security Context  Description | desc | String |  | RW | No | Description of the usage of this OSCOE Security Context. |

|  |  |  |  |
| --- | --- | --- | --- |
| **13.3.2** | **Properties of the Credential Resource** | | |
| **13.3.2.1** | **Credential ID** | | |
| Credential ID ("credid") is a local reference to an entry in a "creds" Property | | | array of the |
| "/oic/sec/cred" Resource. The SRM generates it. The "credid" Property shall | | | be used to |
| disambiguate array elements of the "creds" Property. | | |  |
| **13.3.2.2** | | **Subject UUID** | | |
| The "subjectuuid" Property identifies the Device to which an entry in a "creds" Property array of the | | | | |
| "/oic/sec/cred" Resource shall be used to establish a secure session, verify an authentication | | | | |
| challenge-response or to authenticate an authentication challenge. | | | | |
| A "subjectuuid" Property that matches the Server’s own "deviceuuid" Property, distinguishes the | | | | |
| array entries in the "creds" Property that pertain to this Device. | | | | |
| The "subjectuuid" Property shall be used to identify a group to which a group key is used to protect | | | | |
| shared data. | | | | |
| When certificate chain is used during secure connection establishment, the "subjectuuid" Property | | | | |
| shall also be used to verify the identity of the responder. The presented certificate chain shall be | | | | |
| accepted, if there is a matching Credential entry on the Device that satisfies all of the following: | | | | |
| – Public Data of the entry contains trust anchor (root) of the presented chain. | | | | |
| – Subject UUID of the entry matches UUID in the Common Name field of the End-Entity certificate | | | | |
| in the presented chain. If Subject UUID of the entry is set as a wildcard "\*", this condition is | | | | |
| automatically satisfied. | | | | |
| – Credential Usage of the entry is "oic.sec.cred.trustca". | | | | |
| **13.3.2.3 Role ID** | | | | |
| The "roleid" Property identifies a role that has been granted to the credential. | | | | |
| **13.3.2.4 Credential Type** | | | | |
| The "credtype" Property is used to interpret several of the other Property values whose contents | | | | |
| can differ depending on credential type. These Properties include "publicdata", "privatedata" and | | | | |
| "optionaldata". The "credtype" Property value of "0" ("no security mode") is reserved for testing and | | | | |
| debugging circumstances. Production deployments shall not allow provisioning of credentials of | | | | |
| type "0". The SRM should introduce checking code that prevents its use in production deployments. | | | | |
| **13.3.2.5 Public Data** | | | | |
| The "publicdata" Property contains information that provides additional context surrounding the | | | | |
| issuance of the credential. For example, it might contain information included in a certificate or | | | | |
| response data from a CMS. It might contain wrapped data. | | | | |
| **13.3.2.6 Private Data** | | | | |
| The "privatedata" Property contains secret information that is used to authenticate a Device, protect | | | | |
| data or verify an authentication challenge-response. | | | | |
| The "privatedata" Property shall not be disclosed outside of the SRM’s trusted computing perimeter. | | | | |
| A secure element (SE) or trusted execution environment (TEE) should be used to implement the | | | | |
| SRM’s trusted computing perimeter. The privatedata contents may be referenced using a handle; | | | | |
| for example, if used with a secure storage sub-system. | | | | |
| **13.3.2.7 Optional Data** | | | | |
| The "optionaldata" Property contains information that is optionally supplied, but facilitates key | | | | |
| management, scalability or performance optimization. | | | | |
| **13.3.2.8 Period** | | | | |
| The "period" Property identifies the validity period for the credential. If no validity period is specified, | | | | |
| the credential lifetime is undetermined. Constrained devices that do not implement a date-time | | | | |
| capability shall obtain current date-time information from its CMS. | | | | |

##### 13.3.2.9 Credential Refresh Method Type Definition [Deprecated]

|  |
| --- |
| This clause is intentionally left blank. |
| **13.3.2.10 Credential Usage** |
| Credential Usage indicates to the Device the circumstances in which a credential should be used. |
| Five values are defined: |
| – "oic.sec.cred.trustca": This certificate is a trust anchor for the purposes of certificate chain |
| validation, as defined in [10.4.](#_bookmark173) OCF Server SHALL remove any "/oic/sec/cred" entries with an |
| "oic.sec.cred.trustca" credusage upon transitioning to RFOTM. OCF Servers SHALL use |
| "/oic/sec/cred" entries that have an "oic.sec.cred.trustca" Value of “credusage” Property only |
| as trust anchors for post-onboarding (D)TLS session establishment in RFNOP; these entries |
| are not to be used for onboarding (D)TLS sessions. |
| – "oic.sec.cred.cert": This "credusage" is used for certificates for which the Device possesses the |
| private key and uses it for identity authentication in a secure session, as defined in clause [10.4.](#_bookmark173) |
| – "oic.sec.cred.rolecert": This "credusage" is used for certificates for which the Device possesses |
| the private key and uses to assert one or more roles, as defined in clause [10.4.2.](#_bookmark175) |
| – "oic.sec.cred.mfgtrustca": This certificate is a trust anchor for the purposes of the Manufacturer |
| Certificate Based OTM as defined in clause [7.3.6.](#_bookmark107) OCF Servers SHALL use "/oic/sec/cred" |
| entries that have an "oic.sec.cred.mfgtrustca" Value of "credusage" Property only as trust |
| anchors for onboarding (D)TLS session establishment; these entries are not to be used for post- |
| onboarding (D)TLS sessions. |
| – "oic.sec.cred.mfgcert": This certificate is used for certificates for which the Device possesses |
| the private key and uses it for authentication in the Manufacturer Certificate Based OTM as |
| defined in clause [7.3.6.](#_bookmark107) |
| **13.3.2.11 Resource Owner** |
| The Resource Owner Property allows credential provisioning to occur soon after Device onboarding |
| before access to support services has been established. It identifies the entity authorized to |
| manage the "/oic/sec/cred" Resource in response to Device recovery situations. |
| **13.3.3 Key Formatting** |
| **13.3.3.1 Symmetric Key Formatting** |
| Symmetric keys shall have the format described in [Table 28](#_bookmark233) and [Table 29.](#_bookmark234) |
| **Table 28 – 128-bit symmetric key** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value** | **Type** | **Description** |
| Length | 16 | OCTET | Specifies the number of 8-bit octets following Length |
| Key | opaque | OCTET  Array | 16-byte array of octets. When used as input to a PSK function Length is omitted. |

##### Table 29 – 256-bit symmetric key

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Value** | **Type** | **Description** |
| Length | 32 | OCTET | Specifies the number of 8-bit octets following Length |
| Key | opaque | OCTET  Array | 32-byte array of octets. When used as input to a PSK function Length is omitted. |

* + - 1. **Asymmetric Keys**

Asymmetric key formatting is not available in this revision of the document.

2970

2971

2972

2973

2974

##### Asymmetric Keys with Certificate

Key formatting is defined by certificate definition.

##### Passwords

Password formatting is not available in this revision of the document.

##### 13.3.4 Credential Refresh Method Details [Deprecated]

|  |  |
| --- | --- |
| 2975 | This clause is intentionally left blank. |
| 2976 | **13.4 Certificate Revocation List** |
| 2977 | **13.4.1 CRL Resource Definition [Deprecated]** |
| 2978 | This clause is intentionally left blank. |
| 2979 | **13.5 ACL Resources** |
| 2980 | **13.5.1 ACL Resources General** |
| 2981 | All Resource hosted by a Server are required to match an ACL policy. ACL policies can be |
| 2982 | expressed using "/oic/sec/acl2". The subject (e.g. "deviceuuid" of the Client) requesting access to |
| 2983 | a Resource shall be authenticated prior to applying the ACL check. Resources that are available |
| 2984 | to multiple Clients can be matched using a wildcard subject. All Resources accessible via the |
| 2985 | unsecured communication endpoint shall be matched using a wildcard subject. |
| 2986 | **13.5.2 OCF Access Control List (ACL) BNF defines ACL structures.** |
| 2987 | ACL structure in Backus-Naur Form (BNF) notation is defined in [Table 30:](#_bookmark241) |
| 2988 | **Table 30 – BNF Definition of OCF ACL** |

|  |  |
| --- | --- |
| <ACL> | <ACE> {<ACE>} |
| <ACE> | <SubjectId> <ResourceRef> <Permission> {<Validity>} |
| <SubjectId> | <DeviceId> | <Wildcard> | <RoleId> |
| <DeviceId> | <UUID> |
| <RoleId> | <Character> | <RoleName><Character> |
| <RoleName> | "" | <Authority><Character> |
| <Authority> | <UUID> |
| <ResourceRef> | ' (' <OIC\_LINK> {',' {OIC\_LINK>} ')' |
| <Permission> | ('C' | '-') ('R' | '-') ('U' | '-') ('D' | '-') ('N' | '-') |
| <Validity> | <Period> {<Recurrence>} |
| <Wildcard> | '\*' |
| <URI> | IETF RFC 3986 |
| <UUID> | [IETF RFC 4122](#_bookmark12) |
| <Period> | [IETF RFC 5545](#_bookmark18) Period |
| <Recurrence> | [IETF RFC 5545](#_bookmark18) Recurrence |
| <OIC\_LINK> | [ISO/IEC 30118-1](#_bookmark3) defined in JSON Schema |
| <Character> | <Any UTF8 printable character, excluding NUL> |

|  |
| --- |
| The <DeviceId> token means the requestor must possess a credential that uses <UUID> as its |
| identity in order to match the requestor to the <ACE> policy. |
| The <RoleID> token means the requestor must possess a role credential with <Character> as its |
| role in order to match the requestor to the <ACE> policy. |
| The <Wildcard> token "\*" means any requestor is matched to the <ACE> policy, with or without | |
| authentication. | |
| When a <SubjectId> is matched to an <ACE> policy the <ResourceRef> is used to match the <ACE> | |
| policy to Resources. | |
| The <OIC\_LINK> token contains values used to query existence of hosted Resources. | |
| The <Permission> token specifies the privilege granted by the <ACE> policy given the <SubjectId> | |
| and <ResourceRef> matching does not produce the empty set match. | |
| Permissions are defined in terms of CREATE ("C"), RETRIEVE ("R"), UPDATE ("U"), DELETE ("D"), | |
| NOTIFY ("N") and NIL ("-"). NIL is substituted for a permissions character that signifies the | |
| respective permission is not granted. | |
| The empty set match result defaults to a condition where no access rights are granted. | |
| If the <Validity> token exists, the <Permission> granted is constrained to the time <Period>. | |
| <Validity> may further be segmented into a <Recurrence> pattern where access may alternatively | |
| be granted and rescinded according to the pattern. | |
| **13.5.3 ACL Resource** | |
| An "acl2" is a list of type "ace2". | |
| In order to provide an interface which allows management of array elements of the "aclist2" | |
| Property associated with a "/oic/sec/acl2" Resource, the RETRIEVE, UPDATE and DELETE | |
| operations on the" /oic/sec/acl2" Resource SHALL behave as follows: | |
| 1) A RETRIEVE shall return the full Resource representation. | |
| 2) An UPDATE shall replace or add to the Properties included in the representation sent with the | |
| UPDATE request, as follows: | |
| a) If an UPDATE representation includes the "aclist2" array Property, then: | |
| i) Supplied ACEs with an "aceid" that matches an existing "aceid" shall replace completely | |
| the corresponding ACE in the existing "aclist2" array. | |
| ii) Supplied ACEs without an "aceid" shall be appended to the existing "aclist2" array, and | |
| a unique (to the "/oic/sec/acl2" Resource) "aceid" shall be created and assigned to the | |
| new ACE by the Server. The "aceid" of a deleted ACE should not be reused, to improve | |
| the determinism of the interface and reduce opportunity for race conditions. | |
| iii) Supplied ACEs with an "aceid" that does not match an existing "aceid" shall be | |
| appended to the existing "aclist2" array, using the supplied "aceid". | |
| iv) The rows in [Table 33](#_bookmark245) corresponding to the "aclist2" array Property dictate the Device | |
| States in which an UPDATE of the "aclist2" array Property is always rejected. If OCF | |
| Device is in a Device State where the Access Mode in this row contains "R", then the | |
| OCF Device shall reject all UPDATEs of the "aclist2" array Property. | |
| 3) A DELETE without query parameters shall set the "aclist2" array to the empty array, but shall | |
| not remove the "oic/sec/ace2" Resource. | |
| 4) A DELETE with one or more "aceid" query parameters shall remove the ACE(s) with the | |
| corresponding "aceid"(s) from the "aclist2" array. | |
| 5) The rows in [Table 33](#_bookmark245) corresponding to the "aclist2" array Property dictate the Device States in | |
| which a DELETE is always rejected. If OCF Device is in a Device State where the Access Mode | |
| in this row contains "R", then the OCF Device shall reject all DELETEs. | |
| NOTE The "/oic/sec/acl2" Resource's use of the DELETE operation is not in accordance with the OCF Interfaces | |
| defined in [ISO/IEC 30118-1.](#_bookmark3) | |

Evaluation of local ACL Resource completes when all ACL Resource have been queried and no entry can be found for the requested Resource for the requestor – e.g. "/oic/sec/acl2" does not match the subject and the requested Resource.

[Table 31](#_bookmark243) defines the values of "oic.sec.crudntype".

##### Table 31 – Value Definition of the "oic.sec.crudntype" Property

|  |  |  |  |
| --- | --- | --- | --- |
| **Value** | **Access Policy** | **Description** | **RemarksNotes** |
| bx0000,0000 (0) | No permissions | No permissions | N/A |
| bx0000,0001 (1) | C | CREATE | N/A |
| bx0000,0010 (2) | R | RETREIVE, OBSERVE, DISCOVER | The "R" permission bit covers  both the Read permission and the Observe permission. |
| bx0000,0100 (4) | U | WRITE, UPDATE | N/A |
| bx0000,1000 (8) | D | DELETE | N/A |
| bx0001,0000 (16) | N | NOTIFY | The "N" permission bit is ignored in OCF 1.0, since "R" covers the Observe permission. It is  documented for future versions |

"oic/sec/acl2" Resource is defined in [Table 19.](#_bookmark222)

##### Table 32 – Definition of the "oic/sec/acl2" Resource

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/acl2 | ACL2 | oic.r.acl2 | oic.if.baseli ne, oic.if.rw | Resource for managing access | Security |

[Table 33](#_bookmark245) defines the Properties of "oic.sec.acl2".

##### Table 33 – Properties of the "/oic/sec/acl2" Resource

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Property Name** | **Value Type** | **Mandat ory** | **Device State** | **Access Mode** | **Description** |
| aclist2 | array of  oic.sec.ace2 | Yes | N/A | | The aclist2 Property is an array of ACE  records of type "oic.sec.ace2". The Server uses this list to apply access control to its local Resources. |
| N/A | N/A | N/A | RESET | R | Server shall set to manufacturer defaults. |
| RFOTM | RW | Set by DOTS after successful OTM |
| RFPRO | RW | The AMS (referenced via rowneruuid  property) shall update the aclist entries after mutually authenticated secure session is  established. Access to NCRs is prohibited. |
| RFNOP | R | Access to NCRs is permitted after a matching ACE2 is found. |
| SRESET | RW | The DOTS (referenced via devowneruuid Property of "/oic/sec/doxm Resource")  should evaluate the integrity of and may  update aclist entries when a secure session is established and the Server and DOTS are authenticated. |
| rowneruuid | uuid | Yes | N/A | | The Resource owner Property (rowneruuid) is used by the Server to reference a service provider trusted by the Server. Server shall verify the service provider is authorized to  perform the requested action |
| RESET | R | Server shall set to the nil uuid value (e.g. "00000000-0000-0000-0000-  000000000000" ) |
| RFOTM | RW | The DOTS should configure the rowneruuid Property of "/oic/sec/acl2" Resource when a successful owner transfer session is  established. |
| RFPRO | R | n/a |
| RFNOP | R | n/a |
| SRESET | RW | The DOTS (referenced via devowneruuid Property or rowneruuid Property of  "/oic/sec/doxm" Resource) should verify and if needed, update the Resource owner  Property when a mutually authenticated secure session is established. If the  rowneruuid Property does not refer to a valid  DOTS the Server shall transition to RESET. |

[Table 34](#_bookmark246) defines the Properties of "oic.sec.ace2".

##### Table 34 – "oic.sec.ace2" data type definition.

|  |  |  |  |
| --- | --- | --- | --- |
| **Property Name** | **Value Type** | **Mandatory** | **Description** |
| subject | oic.sec.roletype, oic.sec.didtype,  oic.sec.conntype | Yes | The Client is the subject of the ACE when the roles, Device UUID, or connection type matches. |
| resources | array of  oic.sec.ace2.resource  -ref | Yes | The application’s Resources to which a security policy applies |
| permission | oic.sec.crudntype.bitm ask | Yes | Bitmask encoding of CRUDN permission |
| validity | array of oic.sec.time- pattern | No | An array of a tuple of period and recurrence. Each item in this array contains a string representing a period using the [IETF RFC 5545](#_bookmark18) Period, and a string array representing a recurrence rule using the [IETF RFC 5545](#_bookmark18) Recurrence. |
| aceid | integer | Yes | An aceid is unique with respect to the array entries in the aclist2 Property. |

[Table 35](#_bookmark247) defines the Properties of "oic.sec.ace2.resource-ref ".

##### Table 35 – "oic.sec.ace2.resource-ref" data type definition.

|  |  |  |  |
| --- | --- | --- | --- |
| **Property Name** | **Value Type** | **Manda tory** | **Description** |
| href | uri | No | A URI referring to a Resource to which the containing ACE applies |
| wc | string | No | Refer to [Table 14.](#_bookmark198) |

[Table 36](#_bookmark248) defines the values of "oic.sec.ace2.resource-ref ".

##### Table 36 – Value definition "oic.sec.conntype" Property

|  |  |  |  |
| --- | --- | --- | --- |
| **Property Name** | **Value Type** | **Value Rule** | **Description** |
| conntype | string | enum  [ "auth-crypt",  "anon-clear" ] | This Property allows an ACE to be matched based on the connection or message protection type |
| auth-crypt | ACE applies if the Client is authenticated and the data channel or message is encrypted and integrity protected |
| anon-clear | ACE applies if the Client is not authenticated and the data channel or message is not encrypted but may be integrity protected |

|  |
| --- |
| Local ACL Resources supply policy to a Resource access enforcement point within an OCF stack |
| instance. The OCF framework gates Client access to Server Resources. It evaluates the subject’s |
| request using policies contained in ACL Resources. |
| Resources named in the ACL policy can be fully qualified or partially qualified. Fully qualified |
| Resource references include the device identifier in the href Property that identifies the remote |
| Resource Server that hosts the Resource. Partially qualified references mean that the local |
| Resource Server hosts the Resource. If a fully qualified Resource reference is given, the |
| Intermediary enforcing access shall have a secure channel to the Resource Server and the |
| Resource Server shall verify the Intermediary is authorized to act on its behalf as a Resource |
| access enforcement point. |
| Resource Servers should include references to Device and ACL Resources where access |
| enforcement is to be applied. However, access enforcement logic shall not depend on these |
| references for access control processing as access to Server Resources will have already been |
| granted. |
| Local ACL Resources identify a Resource Owner service that is authorized to instantiate and modify |
| this Resource. This prevents non-terminating dependency on some other ACL Resource. |
| Nevertheless, it should be desirable to grant access rights to ACL Resources using an ACL |
| Resource. |
| An ACE2 entry is considered "currently valid" if the validity period of the ACE2 entry includes the |
| time of the request. The validity period in the ACE2 may be a recurring time period (e.g., daily from |
| 1:00-2:00). Matching the Resource(s) specified in a request to the "resource" Property of the ACE2 |
| is defined in clause [12.2.](#_bookmark194) For example, one way they can match is if the Resource URI in the |
| request exactly matches one of the Resource references in the ACE2 entries. |
| A request will match an ACE2 if any of the following are true: |
| 1) The ACE2 "subject" Property is of type "oic.sec.didtype" has a UUID value that matches the |
| "deviceuuid" Property associated with the secure session; |
| AND the Resource of the request matches one of the "resources" Property of the ACE2 |
| "oic.sec.ace2.resource-ref"; |
| AND the ACE2 is currently valid. |
| 2) The ACE2 "subject" Property is of type "oic.sec.conntype" and has the wildcard value that |
| matches the currently established connection type; |
| AND the Resource of the request matches one of the "resources" Property of the ACE2 |
| "oic.sec.ace2.resource-ref"; |
| AND the ACE2 is currently valid. |
| 3) When Client authentication uses a certificate credential; |
| AND one of the "roleid" values contained in the role certificate matches the "roleid" Property of |
| the ACE2 "oic.sec.roletype"; |
| AND the role certificate public key matches the public key of the certificate used to establish |
| the current secure session; |
| AND the Resource of the request matches one of the array elements of the "resources" |
| Property of the ACE2 "oic.sec.ace2.resource-ref"; |
| AND the ACE2 is currently valid. |
| 4) When Client authentication uses a certificate credential; |
| AND the CoAP payload query string of the request specifies a role, which is member of the set |
| of roles contained in the role certificate; |
| AND the roleid values contained in the role certificate matches the "roleid" Property of the ACE2 |
| "oic.sec.roletype"; |
| AND the role certificate public key matches the public key of the certificate used to establish |
| the current secure session; |
| AND the Resource of the request matches one of the "resources" Property of the ACE2 |
| "oic.sec.ace2.resource-ref"; |
| AND the ACE2 is currently valid. |
| 5) When Client authentication uses a symmetric key credential; |
| AND one of the "roleid" values associated with the symmetric key credential used in the secure |
| session, matches the "roleid" Property of the ACE2 "oic.sec.roletype"; |

|  |  |
| --- | --- |
| 3108 | AND the Resource of the request matches one of the array elements of the "resources" |
| 3109 | Property of the ACE2 "oic.sec.ace2.resource-ref"; |
| 3110 | AND the ACE2 is currently valid. |
| 3111 | 6) When Client authentication uses a symmetric key credential; |
| 3112 | AND the CoAP payload query string of the request specifies a role, which is contained in the |
| 3113 | "oic.r.cred.creds.roleid" Property of the current secure session; |
| 3114 | AND CoAP payload query string of the request specifies a role that matches the "roleid" |
| 3115 | Property of the ACE2 "oic.sec.roletype"; |
| 3116 | AND the Resource of the request matches one of the array elements of the "resources" |
| 3117 | Property of the ACE2 "oic.sec.ace2.resource-ref"; |
| 3118 | AND the ACE2 is currently valid. |
| 3119 | A request is granted if ANY of the ‘matching’ ACE2 entries contain the permission to allow the |
| 3120 | request. Otherwise, the request is denied. |
| 3121 | There is no way for an ACE2 entry to explicitly deny permission to a Resource. Therefore, if one |
| 3122 | Device with a given role should have slightly different permissions than another Device with the |
| 3123 | same role, they must be provisioned with different roles. |
| 3124 | The Server is required to verify that any hosted Resource has authorized access by the Client |
| 3125 | requesting access. The "/oic/sec/acl2" Resource is co-located on the Resource host so that the |
| 3126 | Resource request processing should be applied securely and efficiently. See [Annex A](#_bookmark327) for example. |
| 3127 | **13.6 Access Manager ACL Resource [Deprecated]** |
| 3128 | This clause is intentionally left blank. |
| 3129 | **13.7 Signed ACL Resource [Deprecated]** |
| 3130 | This clause is intentionally left blank. |
| 3131 | **13.8 Provisioning Status Resource** |
| 3132 | The "/oic/sec/pstat" Resource maintains the Device provisioning status. Device provisioning should |
| 3133 | be Client-directed or Server-directed. Client-directed provisioning relies on a Client device to |
| 3134 | determine what, how and when Server Resources should be instantiated and updated. Server- |
| 3135 | directed provisioning relies on the Server to seek provisioning when conditions dictate. Furthermore, |
| 3136 | the "/oic/sec/cred" Resource should be provisioned at ownership transfer with credentials |
| 3137 | necessary to open a secure connection with appropriate support service. |
| 3138 | "/oic/sec/pstat" Resource is defined in [Table 37.](#_bookmark252) |
| 3139 | **Table 37 – Definition of the "/oic/sec/pstat" Resource** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/pstat | Provisioning Status | oic.r.pstat | oic.if.baseline, oic.if.rw | Resource for managing Device provisioning status | Configuration |

[Table 38](#_bookmark253) defines the Properties of "/oic/sec/pstat".

##### Table 38 – Properties of the "/oic/sec/pstat" Resource

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandat ory** | **Access Mode** | **Device State** | **Description** |
| Device Onboarding  State | dos | oic.sec.dostype | N/A | Yes | RW |  | Device Onboarding State |
| Is Device Operational | isop | Boolean | T|F | Yes | R | RESET | Server shall set to FALSE |
| R | RFOTM | Server shall set to FALSE |
| R | RFPRO | Server shall set to FALSE |
| R | RFNOP | Server shall set to TRUE |
| R | SRESET | Server shall set to FALSE |
| Current Mode | cm | oic.sec.dpmtype | bitmask | Yes | R |  | Current Mode |
| Target Mode | tm | oic.sec.dpmtype | bitmask | Yes | RW |  | Target Mode |
| Operational Mode | om | oic.sec.pomtype | bitmask | Yes | R | RESET | Server shall set to manufacturer default. |
| RW | RFOTM | Set by DOTS after successful OTM |
| RW | RFPRO | Set by CMS, AMS, DOTS after successful authentication |
| RW | RFNOP | Set by CMS, AMS, DOTS after successful authentication |
| RW | SRESET | Set by DOTS. |
| Supported Mode | sm | oic.sec.pomtype | bitmask | Yes | R | All states | Supported provisioning services operation modes |
| Device UUID | deviceuui d | String | uuid | Yes | RW | All states | [DEPRECATED] A uuid that  identifies the Device to which the status applies |
| Resource Owner ID | rowneruui d | String | uuid | Yes | R | RESET | Server shall set to the nil uuid value (e.g. "00000000-0000- 0000-0000-000000000000" ) |
| RW | RFOTM | The DOTS should configure the rowneruuid Property when a successful owner transfer session is established. |
| R | RFPRO | n/a |
| R | RFNOP | n/a |
| RW | SRESET | The DOTS (referenced via devowneruuid Property of  "/oic/sec/doxm" Resource) should verify and if needed, update the Resource owner Property when a mutually authenticated secure session is established. If the  rowneruuid does not refer to a valid DOTS the Server shall  transition to RESET. |

[Table 39](#_bookmark254) defines the Properties of "oic.sec.dostype".

##### Table 39 – Properties of the ".oic.sec.dostype" Property

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandator y** | **Access Mode** | **Device State** | **Description** |
| Device Onboarding  State | s | UINT16 | enum (0=RESET, 1=RFOTM,  2=RFPRO,  3=RFNOP,  4=SRESET | Y | R | RESET | The Device is in a hard reset state. |
| RW | RFOTM | Set by DOTS after successful OTM to RFPRO. |
| RW | RFPRO | Set by CMS, AMS, DOTS after successful authentication |
| RW | RFNOP | Set by CMS, AMS, DOTS after successful authentication |
| RW | SRESET | Set by CMS, AMS, DOTS after successful authentication |
| Pending state | p | Boolean | T | F | Y | R | All States | FALSE (0) – "s" state changes are complete. Since Device is not able to respond when the value is TRUE, other values of this property are  DEPRECATED. |

|  |
| --- |
| In all Device states: |
| – The Device permits an authenticated and authorised Client to change the Device state of a |
| Device by updating the "s" Property of the "dos" Property of the "/oic/sec/pstat" Resource to |
| the desired value. The allowed Device state transitions are defined in [Figure 22.](#_bookmark123) |
| – Prior to updating the "s" Property of the "dos" Property of the "/oic/sec/pstat" Resource, the |
| Client configures the Device to meet entry conditions for the new Device state. The SVR |
| definitions define the entity (Client or Server) expected to perform the specific SVR |
| configuration change to meet the entry conditions. Once the Client has configured the aspects |
| for which the Client is responsible, it can update the "s" Property of the "dos" Property of the |
| "/oic/sec/pstat" Resource. The Server then makes any changes for which the Server is |
| responsible, including updating required SVR values, and set the "s" Property of the "dos" |
| Property of the "/oic/sec/pstat" Resource to the new value. |
| When Device state is RESET: |
| – All SVR content is removed and reset to manufacturer default values. |
| – The default manufacturer Device state is RESET. |
| – NCRs are reset to manufacturer default values. |
| – NCRs shall not be accessible. |
| – After successfully processing RESET the SRM transitions to RFOTM by setting the "s" Property |
| of the "dos" Property of the "/oic/sec/pstat" Resource to 1 (RFOTM). |
| When Device state is RFOTM: |
| – NCRs shall not be accessible. |
| – Before OTM is successful, the the "s" Property of the "dos" Property of the "/oic/sec/pstat" |
| Resource is read-only by unauthenticated requestors |
| – After the OTM is successful, the "s" Property of the "dos" Property of the "/oic/sec/pstat" |
| Resource is read-write by authorized requestors. |
| – The negotiated Device OC is used to create an authenticated session over which the DOTS |
| directs the Device state to transition to RFPRO. |
| – If an authenticated session cannot be established the ownership transfer session should be |
| disconnected and SRM sets back the Device state to RESET. |
| – Ownership transfer session, especially Random PIN OTM, should not exceed 60 seconds. If |
| the SRM asserts the OTM failed, the ownership transfer session should be disconnected, and |
| the Device should transition to RESET ("/pstat.dos.s"=0 (RESET)). |
| – The DOTS UPDATES the "devowneruuid" Property in the "/oic/sec/doxm" Resource to a non- |
| nil UUID value. The DOTS (or other authorized client) can update it multiple times while in |
| RFOTM. It is not updatable while in other device states except when the Device state returns |
| to RFOTM through RESET. |
| – The DOTS can have additional provisioning tasks to perform while in RFOTM. When done, the |
| DOTS UPDATES the "owned" Property in the "/oic/sec/doxm" Resource to "true". |
| – After successful OTM, the DOTS triggers the transition to RFPRO and the "s" Property of the |
| "dos" Property of the "/oic/sec/pstat" Resource is set to 2 (RFPRO). |
| When Device state is RFPRO: |
| – The "s" Property of the "dos" Property of the "/oic/sec/pstat" Resource is read-only by |
| unauthorized requestors and read-write by authorized requestors. |
| – NCRs shall not be accessible, except for Easy Setup Resources, if supported. |
| – An authorized Client may provision SVRs as needed for normal functioning in RFNOP. |
| – An authorized Client may perform consistency checks on SVRs to determine which shall be re- |
| provisioned. |
| – Failure to successfully provision SVRs may trigger a state change to RESET. For example, if |
| the Device has already transitioned from SRESET but consistency checks continue to fail. |
| – The authorized Client sets the "s" Property of the "dos" Property of the "/oic/sec/pstat" Resource |
| to 3 (RFNOP). |
| When Device state is RFNOP: |
| – The "s" Property of the "dos" Property of the "/oic/sec/pstat" Resource is read-only by |
| unauthorized requestors and read-write by authorized requestors. |
| – NCRs, SVRs and core Resources are accessible following normal access processing. |
| – When additional provisioning is necessary, the Device may be transitioned to RFPRO by an |
| authorized Client. Only the Device owner should transition to SRESET or RESET. |
| When Device state is SRESET: |
| – NCRs shall not be accessible. The integrity of NCRs may be suspect but the SRM doesn’t |
| attempt to access or reference them. |
| – SVR integrity is not guaranteed, but access to some SVR Properties is necessary. These |
| include "devowneruuid" Property of the "/oic/sec/doxm" Resource, "creds":[{…,{"subjectuuid":<devowneruuid>},…}] Property of the "/oic/sec/cred" Resource and |
| "pstat.dos.s" "/oic/sec/pstat" Resource. |
| – The certificates that identify and authorize the Device owner are sufficient to re-create |
| minimalist "/oic/sec/cred" and "/oic/sec/doxm" Resources enabling Device owner control of |
| SRESET. If the SRM can’t establish these Resources, then it will transition to RESET. |
| – An authorized Client performs SVR consistency checks. The authorized Client can provision |
| SVRs as needed to ensure they are available for continued provisioning in RFPRO or for normal |
| functioning in RFNOP. |
| – The authorized Device owner can avoid entering RESET and RFOTM by UPDATING |
| "pstat.dos.s" with RFPRO or RFNOP values. |
| – ACLs on SVR are presumed to be invalid. Access authorization is granted according to Device |
| owner privileges only. |
| – The SRM asserts a Client-directed operational mode (e.g. "/pstat.om"=4). |
| The provisioning mode type is a 16-bit mask enumerating the various Device provisioning modes. |
| "{ProvisioningMode}" should be used in this document to refer to an instance of a provisioning |
| mode without selecting any particular value. |
| "oic.sec.dpmtype" is defined in [Table 40.](#_bookmark255) |
| **Table 40 – Definition of the "oic.sec.dpmtype" Property** |

|  |  |  |
| --- | --- | --- |
| **Type Name** | **Type URN** | **Description** |
| Device Provisioning Mode | oic.sec.dpmtype | Device provisioning mode is a 16-bit bitmask describing various provisioning modes |

3225

3226

3227

|  |  |
| --- | --- |
| 3228 | The provisioning operation mode type is an 8-bit mask enumerating the various provisioning |
| 3229 | operation modes. |
| 3230 | "oic.sec.pomtype" is defined in [Table 43.](#_bookmark258) |
| 3231 | **Table 43 – Definition of the "oic.sec.pomtype" Property** |

3232

[Table 41](#_bookmark256) and [Table 42](#_bookmark257) define the values of "oic.sec.dpmtype".

##### Table 41 – Value Definition of the "oic.sec.dpmtype" Property (Low-Byte)

|  |  |  |
| --- | --- | --- |
| **Value** | **Device Mode** | **Description** |
| bx0000,0001 (1) | Deprecated |  |
| bx0000,0010 (2) | Deprecated |  |
| bx0000,0100 (4) | Deprecated |  |
| bx0000,1000 (8) | Deprecated |  |
| bx0001,0000 (16) | Deprecated |  |
| bx0010,0000 (32) | Deprecated |  |
| bx0100,0000 (64) | Initiate Software Version Validation | Software version validation requested/pending (1) Software version validation complete (0)  Requires software download to verify integrity of software package |
| bx1000,0000 (128) | Initiate Secure Software Update | Secure software update requested/pending (1) Secure software update complete (0) |

**Table 42 – Value Definition of the "oic.sec.dpmtype" Property (High-Byte)**

|  |  |  |
| --- | --- | --- |
| **Value** | **Device Mode** | **Description** |
| bx0000,0001 (1) | Initiate Software Availability Check | Checks if new software is available on remote endpoint. Does not require to download software.  Methods used are out of bound. |
| Bits 2-8 | <Reserved> | Reserved for later use |

|  |  |  |
| --- | --- | --- |
| **Type Name** | **Type URN** | **Description** |
| Device Provisioning OperationMode | oic.sec.pomtype | Device provisioning operation mode is a 8-bit bitmask describing various provisioning operation modes |

[Table 44](#_bookmark259) defines the values of "oic.sec.pomtype".

3233

|  |  |
| --- | --- |
| 3234 | **13.9 Certificate Signing Request Resource** |
| 3235 | The "/oic/sec/csr" Resource is used by a Device to provide its desired identity, public key to be |
| 3236 | certified, and a proof of possession of the corresponding private key in the form of a IETF RFC |
| 3237 | 2986 PKCS#10 Certification Request. If the Device supports certificates (i.e. the "sct" Property of |
| 3238 | "/oic/sec/doxm" Resource has a 1 in the 0x8 bit position), the Device shall have a "/oic/sec/csr" |
| 3239 | Resource. |
| 3240 | "/oic/sec/csr" Resource is defined in [Table 45.](#_bookmark261) |
| 3241 | **Table 45 – Definition of the "/oic/sec/csr" Resource** |

3242

3243

##### Table 44 – Value Definition of the "oic.sec.pomtype" Property

|  |  |  |
| --- | --- | --- |
| **Value** | **Operation Mode** | **Description** |
| bx0000,0001 (1) | Server-directed utilizing multiple  provisioning services | Deprecated |
| bx0000,0010 (2) | Server-directed utilizing a single  provisioning service | Deprecated |
| bx0000,0100 (4) | Client-directed provisioning | Device supports provisioning service control of this Device’s provisioning operations. This bit is always TRUE. |
| bx0000,1000(8) – bx1000,0000(128) | <Reserved> | Reserved for later use |
| bx1111,11xx | <Reserved> | Reserved for later use |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/csr | Certificate Signing Request | oic.r.csr | oic.if.baseline, oic.if.rw | The CSR Resource contains a Certificate  Signing Request for the Device’s public key. | Configuration |

[Table 46](#_bookmark262) defines the Properties of "/oic/sec/csr ".

##### Table 46 – Properties of the "oic.r.csr" Resource

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Access Mode** | **Mandatory** | **Description** |
| Certificate Signing Request | csr | String | R | Yes | Contains the signed CSR encoded according to the encoding Property |
| Encoding | encoding | String | R | Yes | A string specifying the encoding format of the data contained in the csr Property  "oic.sec.encoding.pem" – Encoding for PEM-encoded certificate signing request |

|  |
| --- |
| The Device chooses which public key to use, and may optionally generate a new key pair for this |
| purpose. |
| In the CSR, the Common Name component of the Subject Name shall contain a string of the format |
| "uuid:X" where X is the Device’s requested UUID in the format defined by [IETF RFC 4122.](#_bookmark12) The |
| Common Name, and other components of the Subject Name, may contain other data. If the Device |
| chooses to include additional information in the Common Name component, it shall delimit it from |
| the UUID field by white space, a comma, or a semicolon. |
| If the Device does not have a pre-provisioned key pair to use, but is capable and willing to generate |
| a new key pair, the Device may begin generation of a key pair as a result of a RETRIEVE of this |

|  |  |
| --- | --- |
| 3253 | Resource. If the Device cannot immediately respond to the RETRIEVE request due to time required |
| 3254 | to generate a key pair, the Device shall return an "operation pending" error. This indicates to the |
| 3255 | Client that the Device is not yet ready to respond, but will be able at a later time. The Client should |
| 3256 | retry the request after a short delay. |
| 3257 | **13.10 Roles Resource** |
| 3258 | The "roles" Resource maintains roles that have been asserted with role certificates, as described |
| 3259 | in clause [10.4.2.](#_bookmark175) Asserted roles have an associated public key, i.e., the public key in the role |
| 3260 | certificate. Servers shall only grant access to the roles information associated with the public key |
| 3261 | of the Client. The roles Resource should be viewed as an extension of the (D)TLS session state. |
| 3262 | See [10.4.2](#_bookmark175) for how role certificates are validated. |
| 3263 | The roles Resource shall be created by the Server upon establishment of a secure (D)TLS session |
| 3264 | with a Client, if is not already created. The roles Resource shall only expose a secured OCF |
| 3265 | Endpoint in the "/oic/res" response. A Server shall retain the roles Resource at least as long as the |
| 3266 | (D)TLS session exists. A Server shall retain each certificate in the roles Resource at least until the |
| 3267 | certificate expires or the (D)TLS session ends, whichever is sooner. The requirements of clause |
| 3268 | 10.3 and [10.4.2](#_bookmark175) to validate a certificate’s time validity at the point of use always apply. A Server |
| 3269 | should regularly inspect the contents of the roles Resource and purge contents based on a policy |
| 3270 | it determines based on its resource constraints. For example, expired certificates, and certificates |
| 3271 | from Clients that have not been heard from for some arbitrary period of time could be candidates |
| 3272 | for purging. |
| 3273 | The OCF namespace ("oic.role.\*") is restricted to OCF-defined roles. "oic.role.owner" is an OCF- |
| 3274 | defined Role that is intended to provide Resource Owner privileges to multiple Clients in a scalable |
| 3275 | way. Servers shall grant access to perform all supported operations in the current Device state |
| 3276 | (see clause [8](#_bookmark121)) on all supported SVRs regardless of ACL configuration the Clients asserting |
| 3277 | "oic.role.owner" Role. Servers shall reject assertion of any Role, which starts with "oic.role.", but |
| 3278 | is not one of the following Roles: |
| 3279 | – "oic.role.owner" |
| 3280 | The "roles" Resource is implicitly created by the Server upon establishment of a (D)TLS session. |
| 3281 | In more detail, the RETRIEVE, UPDATE and DELETE operations on the roles Resource shall |
| 3282 | behave as follows. Unlisted operations are implementation specific and not reliable. |
| 3283 | 1) A RETRIEVE request shall return all previously asserted roles associated with the currently |
| 3284 | connected and authenticated Client’s identity. RETRIEVE requests with a "credid" query |
| 3285 | parameter is not supported; all previously asserted roles associated with the currently |
| 3286 | connected and authenticated Client’s identity are returned. |
| 3287 | 2) An UPDATE request that includes the "roles" Property shall replace or add to the Properties |
| 3288 | included in the array as follows: |
| 3289 | a) If either the "publicdata" or the "optionaldata" are different than the existing entries in the |
| 3290 | "roles" array, the entry shall be added to the "roles" array with a new, unique "credid" value. |
| 3291 | b) If both the "publicdata" and the "optionaldata" match an existing entry in the "roles" array, |
| 3292 | the entry shall be considered to be the same. The Server shall reply with a 2.04 Changed |
| 3293 | response and a duplicate entry shall not be added to the array. |
| 3294 | c) The "credid" Property is optional in an UPDATE request and if included, it may be ignored |
| 3295 | by the Server. The Server shall assign a unique "credid" value for every entry of the "roles" |
| 3296 | array. |
| 3297 | 3) A DELETE request without a "credid" query parameter shall remove all entries from the |
| 3298 | "/oic/sec/roles" Resource array corresponding to the currently connected and authenticated |
| 3299 | Client’s identity. |
| 3300 | 4) A DELETE request with a "credid" query parameter shall remove only the entries of the |
| 3301 | "/oic/sec/roles" Resource array corresponding to the currently connected and authenticated |
| 3302 | Client’s identity and where the corresponding "credid" matches the entry. |
| 3303 | NOTE The "/oic/sec/roles" Resource's use of the DELETE operation is not in accordance with the OCF Interfaces |
| 3304 | defined in [ISO/IEC 30118-1.](#_bookmark3) |
| 3305 | See clause [8](#_bookmark121) for restrictions on the states in which this Resource may be modified. |
| 3306 | "/oic/sec/roles" Resource is defined in [Table 47.](#_bookmark264) |
| 3307 | **Table 47 – Definition of the "/oic/sec/roles" Resource** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/roles | Roles | oic.r.roles | oic.if.basel ine,  oic.if.rw | Resource containing roles that have previously been asserted to this Server | Security |

[Table 48](#_bookmark265) defines the Properties of "/oic/sec/roles".

##### Table 48 – Properties of the "/oic/sec/roles" Resource

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandat ory** | **Description** |
| Roles | roles | oic.sec.cred | array | RW | Yes | List of roles previously asserted to this Server |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3310 | Because "/oic/sec/roles" shares the "oic.sec.cred" schema with "/oic/sec/cred", "subjectuuid" is a required Property. | | | |
| 3311 | However, "subjectuuid" is not used in a role certificate. Therefore, a Device may ignore the "subjectuuid" Property if the | | | |
| 3312 | Property is contained in an UPDATE request to the "/oic/sec/roles" Resource. | | | |
| 3313 | **13.11 Auditable Events List Resource** | | | |
| 3314 | **13.11.1 Auditable Events List Resource General** | | | |
| 3315 | The "/oic/sec/ael" Resource maintains a list of logged Auditable Events. Every OCF Device logs | | | |
| 3316 | AEEs filtered according to the values of the "categoryfilter" and "priorityfilter" Properties of | | | |
| 3317 | "/oic/sec/ael" Resource. All Devices shall have a "/oic/sec/ael" Resource to maintain AEEs. The | | | |
| 3318 | new AEE shall be added to the "events" Property of "/oic/sec/ael" Resource as the last entry in the | | | |
| 3319 | array. A Device shall store all AEEs of the "/oic/sec/ael" Resource in non-volatile memory. A Device | | | |
| 3320 | shall be able to store at least 1 AEE. | | | |
| 3321 | The "categoryfilter" Property determines what categories of AEEs are to be logged. The | | | |
| 3322 | "categoryfilter" Property is an integer value which is a composition of bitmasks. A Device shall log | | | |
| 3323 | all AEEs filtered by this value. If the "categoryfilter" is either set to 0xff or is not set, then the Device | | | |
| 3324 | shall log AEEs of all categories. Refer to [Table 50](#_bookmark269) for more details. | | | |
| 3325 | The "priorityfilter" Property determines the lowest priority of AEE to be logged. A smaller value | | | |
| 3326 | means higher priority. The AEEs whose "priority" Property values are equal to or smaller than this | | | |
| 3327 | value shall be logged. If the "priorityfilter" Property is either set to the highest priority or is not set, | | | |
| 3328 | then the Device shall log all AEEs. No matter what value is set to "priorityfilter", an AEE of CRIT | | | |
| 3329 | (== 0) "priority" shall always be logged. Refer to [Table 50](#_bookmark269) for more details. | | | |
| 3330 | When an AEE is added, the "usedspace" Property shall be updated to reflect the total storage used | | | |
| 3331 | by all logged events. When the reserved storage for AEEs is full, the oldest AEE shall be purged. | | | |
| 3332 | A Device logs a new AEE as follows: | | | |
| 3333 | 5) If a new AEE is not filtered by "categoryfilter" and "priorityfilter", then it is dropped. | | | |
| 3334 | /\* c-like pseudo code \*/ | | | |
| 3335 | If ((categoryfilter & new\_aee->category) && (priorityfilter >= new\_aee->priority)) | | | |
| 3336 | | { |  |
| 3337 | |  | addAEE(new\_aee); |
| 3338 | | } |  |
| 3339 | | else |  |
| 3340 | | { |  |
| 3341 | |  | free(new\_aee); |

}

1. If the value of "usedspace" Property is equal to, or the sum of the "usedspace" Property value and the size of the new AEE is bigger than the value of the "maxspace" Property of "/oic/sec/ael" Resource, then:
   1. Remove the oldest AEE continuously while the sum of the "usedspace" Property value and the size of the new AEE is bigger than the "maxspace" Property value.

/\* c-like pseudo code \*/

Int addAEE(AEEtype \*new\_aee)

{

While ((usespace + new\_aee->size) > maxspace)

{

/\* purgeAEE() returns the size of purged AEE \*/ sizeOfPurgedAEE = purgeAEE();

usedspace -= sizeOfPurgedAEE;

}

...

...

}

1. Add the new AEE to the "events" array Property of the "/oic/sec/ael" Resource as the last entry in the array.
2. Increase the value of the "usedspace" Property by the size of the new AEE.

In order to provide a mechanism which allows management of the "events" array Property, the RETRIEVE and UPDATE operations on the "/oic/sec/ael" Resource shall behave as follows:

1. A RETRIEVE operation shall return the full Resource representation.
2. An UPDATE operation may set the "categoryfilter" and/or "priorityfilter" Properties. The "/oic/sec/ael" Resource is defined in [Table 49.](#_bookmark268)

##### Table 49 – Definition of the "/oic/sec/ael" Resource

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt"**  **value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/ael | Auditable Event List | oic.r.ael | oic.if.baseline, oic.if.rw | Resource for storing AEEs | Security |

|  |  |
| --- | --- |
| 3371 |  |
| 3372 | [Table 50](#_bookmark269) defines the Properties of the "/oic/sec/ael" Resource. |
| 3373 | **Table 50 – Properties of the "/oic/sec/ael" Resource** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Man dato ry** | **Device State** | **Acc ess Mo de** | **Description** |
| AEE list | "events" | "array" |  | Yes | RESET | R | The Device clears |
|  |  |  | Array of "oic.sec.aee"  entries |  | RFOTM | R | This list stores AEEs whose "category"  Property value is filtered by  "categoryfilter" Property and "priority" Property value is equal or less than the value of "priorityfilter" Property. |
| RFPRO |
| RFNOP |
| SRESET |
| current used storage size | "usedspace" | "integer" | >= 0  (default: 0) | Yes | RESET | R | The Device sets to 0 |
| RFOTM | R | Current used space for logged AEEs.  The Device updates this Property whenever new AEEs are logged. |
| RFPRO |
| RFNOP |
| SRESET |
| maximum allowed  storage size for AEEs | "maxspace" | "integer" | > 0 | Yes |  | R | This means the maximum allowable storage size for AEEs that can be stored in "events" list. The  Manufacturer chooses this value. |
|  |  |  | enum |  |  |  |  |
| unit for  storage size | "unit" | "string" | ["Kbyte",  "Byte"]  (default: “Byte”) | No | R | The unit for "usedspace" and "maxspace" Properties. The  Manufacturer chooses this value. |
|  |  |  |  |  | RESET | R | The Device sets to the manufacturer default value |
|  |  |  |  |  | RFOTM |  | This value decides what categories of AEEs are to be logged. |
|  |  |  |  |  |  | RW | Meaning of each bit:   * 0x01 (Access Control) |
|  |
|  |  |  |  |  | RFPRO |  | * 0x02 (Onboarding) |
| Categories of AEE to be logged | "categoryfilter" | "integer" | bitmask  (default: 0xff) | Yes |  |  | * 0x04 (Device) * 0x08 (Authentication) * 0x10 (SVR Modification) |
| RFNOP | R |
|  |  |  |  |  |  |  | * 0x20 (Cloud) |
|  |  |
|  |  |  |  |  |  |  | * 0x40 (Communication) |
|  |  |  |  |  |  |  | * 0x80 (Reserved) |
|  |  |  |  |  | SRESET | RW | e.g.) if "categoryfilter" == 0xff: log all events of all categories |
|  |  |  |  |  |  |  | e.g.) if "categoryfilter" == 0x03: log all events of 'AC (== 0x01)' and 'OB (==0x02)' categories |
|  |  |  |  |  | RESET | R | Device sets to manufacturer default value |
| Minimum priority of AEEs to be  logged | "priorityfilter" | "integer" | enum  [0, 1, 2, 3, 4]  (default: 4) | Yes | RFOTM | RW | The AEEs whose "priority" values are equal to or smaller than this value are  logged. A smaller value means a higher priority.  Meaning of each value:   * 0 (CRIT) * 1 (ERR) |
| RFPRO |
|  |  |
|  |  |  |  |  | RFNOP | R | * 2 (WARN) |
|  |  |  |  |  |  |  | * 3 (INFO) |
|  |  |  |  |  |  |  | * 4 (DEBUG) |
|  |  |  |  |  | SRESET | RW | e.g.) if "priorityfilter" is set to DEBUG (==4) all AEEs will be logged |
|  |  |  |  |  |  |  | e.g.) if "priorityfilter" is set to 1, CRIT (==0) and ERR (==1) SEEs will be  logged |

[Table 51](#_bookmark270) defines the Properties of the "oic.sec.aee" type.

##### Table 51 – "oic.sec.aee" data type definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Acce ss Mode** | **Manda tory** | **Devi ce Stat e** | **Description** |
| Auditable Event  Identifier | "aeid" | "string" | N/A | R | Yes | - | Identity of the logged event |
| Category of AEE | "category" | "integer" | enum  [1, 2, 4, 8,  16, 32, 64,  128] | R | Yes | - | The category of this AEE:   * 0x01 (Access Control) * 0x02 (Onboarding) * 0x04 (Device) * 0x08 (Authentication) * 0x10 (SVR Modification) * 0x20 (Cloud) * 0x40 (Communication) * 0x80 (Reserved) |
| Priority of AEE | "priority" | "integer" | enum  [0, 1, 2, 3,  4] | R | Yes | - | The priority of this AEE:   * 0 (CRIT) * 1 (ERR) * 2 (WARN) * 3 (INFO) * 4 (DEBUG) |
| Time stamp | "timestam p" | "string" | date-time (RFC3339  clause 5.6) | R | Yes | - | The time when the AEE occured |
| Event message | "message" | "string" | N/A | R | Yes | - | The description of the logged AEE. |
| Auxiliary info | "auxiliaryi nfo" | "array" | Array of strings | R | Yes | - | Supplementary  information for the "message" Property  e.g.) URI of specific Resource in ACE2 |

|  |
| --- |
| OCF-defined AEEs are listed in [Table 53,](#_bookmark278) and each such AEE has its own values for the "category" |
| and "priority" Properties. |
| The "timestamp" Property follows a full-date and partial-time format of RFC3339. Every new AEE |
| shall have a later timestamp than the latest previously logged AEE. |
| The "auxiliaryinfo" Property provides supplementary info which is not covered by the description in |
| "message" Property. For example, the URI of specific Resource in ACE2 could be "auxiliaryinfo" |
| for "Access Denied" AEE. Please see [Table 53](#_bookmark278) "List of Auditable Events". |
| **13.12****Security Virtual Resources (SVRs) and Access Policy** |
| The SVRs expose the security-related Properties of the Device. |
| Granting access requests (RETRIEVE, UPDATE, DELETE, etc.) for these SVRs to unauthenticated |
| (anonymous) Clients could create privacy or security concerns. |
| For example, when the Device onboarding State is RFOTM, it is necessary to grant requests for |
| the "/oic/sec/doxm" Resource to anonymous requesters, so that the Device can be discovered and |
| onboarded by an OBT. Subsequently, it might be preferable to deny requests for the |
| "/oic/sec/doxm" Resource to anonymous requesters, to preserve privacy. |
| **13.13 SVRs, Discoverability and OCF Endpoints** |
| All implemented SVRs shall be "discoverable" (reference [ISO/IEC 30118-1,](#_bookmark3) Policy Parameter |
| clause 7.8.2.1.2). |
| All implemented discoverable SVRs shall expose a Secure OCF Endpoint (e.g. CoAPS) (reference |
| [ISO/IEC 30118-1,](#_bookmark3) clause 10). |
| The "/oic/sec/doxm" Resource shall expose an Unsecure OCF Endpoint (e.g. CoAP) in RFOTM |
| (reference [ISO/IEC 30118-1,](#_bookmark3) clause 10). |
| **13.14 Additional Privacy Consideration for Core Resources** |
| Unique immutable identifiers are a privacy consideration due to their potential for being used as a |
| tracking mechanism. These include the following Resources and Properties: |
| – "/oic/d" Resource containing the "piid" Property. |
| – "/oic/p" Resource containing the "pi" Property. |
| These identifiers are unique values that are visible at various times throughout the Device lifecycle |
| by anonymous requestors. This implies any Client Device, including those with malicious intent, |
| are able to reliably obtain identifiers useful for building a log of activity correlated with a specific |
| Platform and Device. |
| The "di" Property in the "/oic/d" Resource shall mirror that of the "deviceuuid" Property of the |
| "/oic/sec/doxm" Resource. The DOTS should provision an ACL policy that restricts access to the |
| "/oic/d" Resource such that only authenticated Clients are able to obtain the "di" Property of "/oic/d" |
| Resource. See clause 13.1 for deviceuuid Property lifecycle requirements. |
| Servers should expose a temporary, non-repeated, "piid" Property of "/oic/d" Resource Value upon |
| entering RESET. Servers shall expose a persistent value via the "piid" Property of "/oic/d" Property |
| when the DOTS sets "devowneruuid" Property to a non-nil-UUID value. The DOTS should provision |
| an ACL policy on the "/oic/d" Resource such that only authenticated Clients are able to obtain the |
| "piid" Property of "/oic/d" Resource |
| Servers should expose a temporary, non-repeated, "pi" Property value upon entering RESET. |
| Servers shall expose a persistent value via the "pi" Property of the "/oic/p" Resource when the |
| DOTS sets "devowneruuid" Property to a non-nil-UUID value. The DOTS should provision an ACL |
| policy on the "/oic/p" Resource such that only authenticated Clients are able to obtain the "pi" |
| Property. |
| [Table 52](#_bookmark274) depicts Core Resource Properties Access Modes given various Device States. |
| **Table 52 – Core Resource Properties Access Modes given various Device States** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Resource Type** | **Property title** | **Prope rty**  **name** | **Value type** | **Access Mode** | | **Behaviour** |
| oic.wk.p | Platform ID | pi | oic.types- schema.uuid | All States | R | Server exposes a temporary random UUID when in RESET. |
| oic.wk.d | Permanent  Immutable ID | piid | oic.types- schema.uuid | All States | R | Server exposes a temporary random UUID when in RESET. |
| oic.wk.d | Device Identifier | di | oic.types- schema.uuid | All states | R | /d di mirrors the value contained in "/doxm" "deviceuuid" in all device states. |

|  |  |
| --- | --- |
| 3423 | **13.15 Easy Setup Resource Device State** |
| 3424 | This clause only applies to a new Device that uses Easy Setup for ownership transfer as defined |
| 3425 | in [OCF Wi-Fi Easy Setup.](#_bookmark5) Easy Setup has no impact to new Devices that have a different way of |
| 3426 | connecting to the network i.e. DOTS and AMS don't use a Soft AP to connect to non-Easy Setup |
| 3427 | Devices. |
| 3428 | [Figure 29](#_bookmark276) shows an example of Soft AP and Easy Setup Resource in different Device states. |

**RESET**

- Soft AP Initialized

-/EasySetupResURI Resource inaccessible

**RFOTM**

* DOTS connected to Device via Soft AP
* DOTS on-boards the Device

-/EasySetupResURI Resource inaccessible

3429

**RFNOP**

- Soft AP is disabled when New Device connects with the regular network.

-Discovery & UPDATE of

/EasySetupResURI Resources

3430

3431

3432

3433

3434

3435

3436

3437

3438

3439

3440

3441

3442

3443

3444

3445

3446

##### Figure 29 – Example of Soft AP and Easy Setup Resource in different Device states

**RFPRO**

* AMS connected to Device via Soft AP
* AMS configure ACE2 to allow only Mediator to configure

/EasySetupResURI

-Discovery & UPDATE of

/EasySetupResURI Resources

-SoftAP and AMS disconnected

Device enters RFOTM, Soft AP may be accessible in RFOTM and RFPRO.

While it is reasonable for an End User to expect that power cycling a new Device will turn on the Soft AP for Easy Setup during the initial setup, since that is potentially how it behaved on first boot, it is a security risk to make this the default behaviour of a device that remains unenrolled beyond a reasonable period after first boot.

Therefore, the Soft AP for Easy Setup has several requirements to improve security:

* Time availability of Easy Setup Soft AP should be minimised, and shall not exceed 30 minutes after Device factory reset, RESET or first power boot, or when an End User initiates the Soft AP for Easy Setup.
* If a new Device tried and failed to complete Easy Setup Enrolment immediately following the first boot, or after a factory reset, it may turn the Easy Setup Soft AP back on automatically for another 30 minutes upon being power cycled, provided that the power cycle occurs within 3 hours of first boot or the most recent factory reset. If the End User has initiated the Easy Setup Soft AP directly without a factory reset, it is not necessary to turn it back on if it was on immediately prior to power cycle, because the End User obviously knows how to initiate the process manually.

|  |
| --- |
| – After 3 hours from first boot or factory reset without successfully enrolling the device, the Soft |
| AP should not turn back on for Easy Setup until another factory reset occurs, or the End User |
| initiates the Easy Setup Soft AP directly. |
| – Easy Setup Soft AP may stay enabled during RFNOP, until the Mediator instructs the new |
| Device to connect to the Enroller. |
| – The Easy Setup Soft AP shall be disabled when the new Device successfully connects to the |
| Enroller. |
| – Once a new Device has successfully connected to the Enroller, it shall not turn the Easy Setup |
| Soft AP back on for Easy Setup Enrolment again unless the Device is factory reset, or the End |
| User initiates the Easy Setup Soft AP directly. |
| – Just Works OTM shall not be enabled on Devices which support Easy Setup. |
| – The Soft AP shall be secured (e.g. shall not expose an open AP). |
| – The Soft AP shall support a passphrase for connection by the Mediator, and the passphrase |
| shall be between and 8 and 64 ASCII printable characters. The passphrase may be printed on |
| a label, sticker, packaging etc., and may be entered by the End User into the Mediator device. |
| – The Soft AP should not use a common passphrase across multiple Devices. Instead, the |
| passphrase may be sufficiently unique per device, to prevent guessing of the passphrase by an |
| attacker with knowledge of the Device type, model, manufacturer, or any other information |
| discoverable through Device’s exposed interfaces. |
| The Enrollee shall support WPA2 security (i.e. shall list WPA2 in the "swat" Property of the |
| "/example/WiFiConfResURI" Resource), for potential selection by the Mediator in connecting the |
| Enrollee to the Enroller. The Mediator should select the best security available on the Enroller, for |
| use in connecting the Enrollee to the Enroller. |
| The Enrollee may not expose any interfaces (e.g. web server, debug port, NCRs, etc.) over the |
| Soft AP, other than SVRs, and Resources required for Wi-Fi Easy Setup. |
| The "/example/EasySetupResURI" Resource should not be discoverable in RFOTM or SRESET. |
| After ownership transfer process is completed with the DOTS, and the Device enters in RFPRO, |
| the "/example/EasySetupResURI" may be Discoverable. |
| The OTM CoAPS session may be used by Mediator for connection over Soft AP for ownership |
| transfer and initial Easy Setup provisioning. SoftAP or regular network connection may be used by |
| AMS for "/oic/sec/acl2" Resource provisioning in RFPRO. The CoAPS session authentication and |
| encryption is already defined in the Security spec. |
| In RFPRO, AMS is expected to configure ACL2 Resource on the Device with ACE2 for following |
| Resources to be only configurable by the Mediator with permission to UPDATE or RETRIEVE |
| access: |
| – "/example/EasySetupResURI" |
| – "/example/WifiConfResURI" |
| – "/example/DevConfResURI" |
| An ACE2 granting RETRIEVE or UPDATE access to the Easy Setup Resource |
| { |
| "subject": { "uuid": "<insert-UUID-of-Mediator>" }, |
| "resources": [ |
| { "href": "/example/EasySetupResURI" }, |
| { "href": "/example/WiFiConfResURI" }, |
| { "href": "/example/DevConfResURI" }, |
| ], |

"permission": 6 // RETRIEVE (2) or UPDATE and RETRIEVE(6)

}

|  |  |
| --- | --- |
| 3495 | ACE2 may be re-configured after Easy Setup process. These ACE2s should be installed prior to |
| 3496 | the Mediator performing any RETRIEVE/UPDATE operations on these Resources. |
| 3497 | In RFPRO or RFNOP, the Mediator should discover /EasySetupResURI Resources and UPDATE |
| 3498 | these Resources. The Mediator may UPDATE /EasySetupResURI Resources in RFNOP Device |
| 3499 | state. |
| 3500 | A Mediator shall be hosted on an OCF Device. |
| 3501 | **13.16 List of Auditable Events** |
| 3502 | Whenever a Device detects an occurrence of any of the Auditable Events in [Table 53,](#_bookmark278) then the |
| 3503 | Device shall log an AEE using the corresponding "category", "priority" and "auxiliaryinfo" Properties |
| 3504 | defined in [Table 53.](#_bookmark278) The "auxiliaryinfo" Property shall contain the entries in the "auxiliaryinfo" |
| 3505 | column of [Table 53](#_bookmark278) in the order specified in the table with each bullet contained in a separate array |
| 3506 | entry. The "auxiliaryinfo" Property may contain additional entries for further information following |
| 3507 | the entries for mandatory information. The "aeid" Property shall include the corresponding |
| 3508 | Auditable Event Identifier from [Table 53.](#_bookmark278) |
| 3509 | **Table 53 – List of mandatory Auditable Events and corresponding Property values** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Auditable Event**  **Identifier ("aeid")** | **Auditable Event Description** | **Example "message"** | **"categ ory"** | **"priority"** | **"auxiliaryinfo"** |
| **AC-1** | A Device received a request from an  authenticated Client with valid URI path, valid interface and valid operation for that Resource, but for  which access was denied. | “Access Denied” | 0x01  (Acces s  Control  ) | 2 (WARN) | * Client IP address & port in format [xxxx:..:xxxx]:xxxx * Client UUID in UUID format (e.g. "00000000-0000-0000- 0000-000000000000") * Resource URI (e.g. "/oic/sec/ael") * Requested CRUDN operation (e.g. "CREATE") * Server security state (e.g. "RFNOP") * Asserted roles by Client (e.g. "oic.role.owner"), or "No roles asserted" if there are none |
| **AUTH-1** | The Device  encountered an error during a DTLS  handshaking procedure due to a credential  validation failure. | “DTLS  handshake failed due to a credential  validation failure” | 0x08  (Authen tication  ) | 1 (ERR) | * Client IP address & port in format [xxxx:..:xxxx]:xxxx   ~~~~ |
| **COMM-1** | The Device received a CoAP request which contained unexpected  /unsupported CoAP  header parameters or  unexpected/unsupporte d CoAP options. | “Unexpected CoAP  Command” | 0x40 (COMM  ) | 2 (WARN) | * Client IP address & port in format [xxxx:..:xxxx]:xxxx * Hex-encoded CoAP header in format [xx:xx:xx:xx] * Hex-encoded CoAP options except payload (empty if not present) |

|  |  |
| --- | --- |
| 3510 | Whenever a Device detects an occurrence of any of the Auditable Events in [Table 54,](#_bookmark279) then the |
| 3511 | Device should log an AEE using the corresponding "category", "priority" and "auxiliaryinfo" |
| 3512 | Properties defined in [Table 54.](#_bookmark279) The "auxiliaryinfo" Property shall contain the entries in the |
| 3513 | "auxiliaryinfo" column of [Table 54](#_bookmark279) in the order specified in the table with each bullet contained in a |
| 3514 | separate array entry. The "auxiliaryinfo" Property may contain additional entries for further |

information following the entries for mandatory information. The "aeid" Property shall include the corresponding Auditable Event Identifier from [Table 54.](#_bookmark279)

##### Table 54 – List of recommended Auditable Events and corresponding Property values

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Auditable Event**  **Identifier** | **Auditable Event Description** | **Example "message"** | **"category"** | **"priority"** | **"auxiliaryinfo"** |
| **SVR-1** | The Device's attempted to use one of its  credentials, and detected that the  credential is expired | “My credential is expired” | 0x10  (SVR Modification) | 2 (WARN) | * credid * Credential   expiration value |
| **SVR-2** | The Device could not validate the role  certificate being asserted | "Role assertion failed" | 0x10  (SVR Modification) | 2 (WARN) | * Client IP address & port in format [xxxx:..:xxxx]:xxx x |

|  |  |
| --- | --- |
| 3519 | **13.17 Security Domain Information Resource** |
| 3520 | The "/oic/sec/sdi" Resource contains the information that identifies the OCF Security Domain to |
| 3521 | which the Device belongs. OCF Security Domains are uniquely identifiable. |
| 3522 | This Resource is optional to implement. When it is exposed by a Device, an OCF Onboarding Tool |
| 3523 | (OBT) is expected to provision a random UUID and a Security Domain Name for the OCF Security |
| 3524 | Domain. These two fields are provisioned to a Device during the onboarding process. |
| 3525 | "oic.r.sdi" Resource Type is defined in [Table 55.](#_bookmark281) |
| 3526 | **Table 55 –Definition of the "oic.r.sdi" Resource Type** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF Interfaces** | **Description** | **Related Functional Interaction** |
| "/oic/sec/sdi" | Security Domain Information | "oic.r.sdi" | "oic.if.baseline" "oic.if.rw" | Resource containing Security Domain  information | Configuration |

3527

3528

[Table 56](#_bookmark282) defines the Properties of "oic.r.sdi".

##### Table 56 – Properties of the "oic.r.sdi" Resource Type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Mandat ory** | **Access Mode** | **Device State** | **Description** |
| Security Domain UUID | "uuid" | string | "uuid" | Yes | R | RESET | A UUID that identifies the  Security Domain, set by DOTS during onboarding. |
| RW | RFOTM |
| R | RFPRO |
| R | RFNOP |
| R | SRESET |
| Security Domain Name | "name" | string | N/A | Yes | R | RESET | Human-friendly name for the Security Domain, set by DOTS during onboarding. |
| RW | RFOTM |
| RW | RFPRO |
| R | RFNOP |
| RW | SRESET |
| Privacy Flag | "priv" | boolean | N/A | Yes | R | RESET | Flag to indicate whether the Security Domain Information is copied to "/oic/res", and thus whether it is publicly visible or private. |
| RW | RFOTM |
| RW | RFPRO |
| R | RFNOP |
| RW | SRESET |

|  |  |
| --- | --- |
| 3529 | The purpose of the "priv" Property is to control whether information about a Device’s OCF Security |
| 3530 | Domain is exposed during multicast discoveries. |
| 3531 | If the "priv" Property is set to "false", then the "/oic/res" Resource shall expose its "sduuid" and |
| 3532 | "sdname" Properties with values copied from the "uuid" and "name" Properties of the “/oic/sec/sdi” |
| 3533 | Resource, respectively. |
| 3534 | If the "priv" Property is set to "true", then the "/oic/res" Resource shall not expose its "sduuid" and |
| 3535 | "sdname" Properties. |

|  |  |
| --- | --- |
| 3536 | **14 Security Hardening Guidelines/ Execution Environment Security** |
| 3537 | **14.1 Preamble** |
| 3538 | This is an informative clause. Many TGs in OCF have security considerations for their protocols |
| 3539 | and environments. These security considerations are addressed through security mechanisms |
| 3540 | specified in the security documents for OCF. However, effectiveness of these mechanisms depends |
| 3541 | on security robustness of the underlying hardware and software Platform. This clause defines the |
| 3542 | components required for execution environment security. |
| 3543 | **14.2 Execution Environment Elements** |
| 3544 | **14.2.1 Execution Environment Elements General** |
| 3545 | Execution environment within a computing Device has many components. To perform security |
| 3546 | functions in a robustness manner, each of these components has to be secured as a separate |
| 3547 | dimension. For instance, an execution environment performing AES cannot be considered secure |
| 3548 | if the input path entering keys into the execution engine is not secured, even though the partitions |
| 3549 | of the CPU, performing the AES encryption, operate in isolation from other processes. Different |
| 3550 | dimensions referred to as elements of the execution environment are listed below. To qualify as a |
| 3551 | secure execution environment (SEE), the corresponding SEE element must qualify as secure. |
| 3552 | – (Secure) Storage |
| 3553 | – (Secure) Execution engine |
| 3554 | – (Trusted) Input/output paths |
| 3555 | – (Secure) Time Source/clock |
| 3556 | – (Random) number generator |
| 3557 | – (Approved) cryptographic algorithms |
| 3558 | – Hardware Tamper (protection) |
| 3559 | NOTE Software security practices (such as those covered by Open Web Application Security Project) are outside |
| 3560 | scope of this document, as development of secure code is a practice to be followed by the open source development |
| 3561 | community. This document will however address the underlying Platform assistance required for executing software. |
| 3562 | Examples are secure boot and secure software upgrade. |
| 3563 | Each of the elements above are described in the clauses [14.2.2,](#_bookmark287) [14.2.3,](#_bookmark290) [14.2.4,](#_bookmark291) [14.2.5,](#_bookmark292) [14.2.6,](#_bookmark293) |
| 3564 | [14.2.7.](#_bookmark295) |
| 3565 | **14.2.2 Secure Storage** |
| 3566 | **14.2.2.1 Secure Storage General** |
| 3567 | Secure storage refers to the physical method of housing sensitive or confidential data ("Sensitive |
| 3568 | Data"). Such data could include but not be limited to symmetric or asymmetric private keys, |
| 3569 | certificate data, OCF Security Domain access credentials, or personal user information. Sensitive |
| 3570 | Data requires that its integrity be maintained, whereas Critical Sensitive Data requires that both its |
| 3571 | integrity and confidentiality be maintained. |
| 3572 | It is strongly recommended that IoT Device makers provide reasonable protection for Sensitive |
| 3573 | Data so that it cannot be accessed by unauthorized Devices, groups or individuals for either |
| 3574 | malicious or benign purposes. In addition, since Sensitive Data is often used for authentication and |
| 3575 | encryption, it must maintain its integrity against intentional or accidental alteration. |
| 3576 | A partial list of Sensitive Data is outlined in [Table 57:](#_bookmark288) |

3577

##### Table 57 – Examples of Sensitive Data

|  |  |  |
| --- | --- | --- |
| **Data** | **Integrity protection** | **Confidentiality protection** |
| Owner PSK (Symmetric Keys) | Yes | Yes |
| Service provisioning keys | Yes | Yes |
| Asymmetric Private Keys | Yes | Yes |
| Certificate Data and Signed Hashes | Yes | Not required |
| Public Keys | Yes | Not required |
| Access credentials (e.g. SSID, passwords, etc.) | Yes | Yes |
| ECDH/ECDH Dynamic Shared Key | Yes | Yes |
| Root CA Public Keys | Yes | Not required |
| Device and Platform IDs | Yes | Not required |
| Easy Setup Resources | Yes | Yes |
|  |  |  |
|  |  |  |
| Access Token | Yes | Yes |

|  |  |
| --- | --- |
| 3578 | Exact method of protection for secure storage is implementation specific, but typically combinations |
| 3579 | of hardware and software methods are used. |
| 3580 | **14.2.2.2 Hardware Secure Storage** |
| 3581 | Hardware secure storage is recommended for use with critical Sensitive Data such as symmetric |
| 3582 | and asymmetric private keys, access credentials, and personal private data. Hardware secure |
| 3583 | storage most often involves semiconductor-based non-volatile memory ("NVRAM") and includes |
| 3584 | countermeasures for protecting against unauthorized access to Critical Sensitive Data. |
| 3585 | Hardware-based secure storage not only stores Sensitive Data in NVRAM, but also provides |
| 3586 | protection mechanisms to prevent the retrieval of Sensitive Data through physical and/or electronic |
| 3587 | attacks. It is not necessary to prevent the attacks themselves, but an attempted attack should not |
| 3588 | result in an unauthorized entity successfully retrieving Sensitive Data. |
| 3589 | Protection mechanisms should provide JIL Moderate protection against access to Sensitive Data |
| 3590 | from attacks that include but are not limited to: |
| 3591 | 1) Physical decapping of chip packages to optically read NVRAM contents |
| 3592 | 2) Physical probing of decapped chip packages to electronically read NVRAM contents |
| 3593 | 3) Probing of power lines or RF emissions to monitor voltage fluctuations to discern the bit patterns |
| 3594 | of Critical Sensitive Data |
| 3595 | 4) Use of malicious software or firmware to read memory contents at rest or in transit within a |
| 3596 | microcontroller |
| 3597 | 5) Injection of faults that induce improper Device operation or loss or alteration of Sensitive Data |
| 3598 | **14.2.2.3 Software Storage** |
| 3599 | It is generally NOT recommended to rely solely on software and unsecured memory to store |
| 3600 | Sensitive Data even if it is encrypted. Critical Sensitive Data such as authentication and encryption |
| 3601 | keys should be housed in hardware secure storage whenever possible. |
| 3602 | Sensitive Data stored in volatile and non-volatile memory shall be encrypted using acceptable |
| 3603 | algorithms to prevent access by unauthorized parties through methods described in [14.2.2.2.](#_bookmark289) |

3604

##### 14.2.2.4 Additional Security Guidelines and Best Practices

|  |  |
| --- | --- |
| 3605 | Some general practices that can help ensure that Sensitive Data is not compromised by various |
| 3606 | forms of security attacks: |
| 3607 | 1) FIPS Random Number Generator ("RNG") – Insufficient randomness or entropy in the RNG |
| 3608 | used for authentication challenges can substantially degrade security strength. For this reason, |
| 3609 | it is recommended that a FIPS 800-90A-compliant RNG with a certified noise source be used |
| 3610 | for all authentication challenges. |
| 3611 | 2) Secure download and boot – To prevent the loading and execution of malicious software, where |
| 3612 | it is practical, it is recommended that Secure Download and Secure Boot methods that |
| 3613 | authenticate a binary’s source as well as its contents be used. |
| 3614 | 3) Deprecated algorithms – Algorithms included but not limited to the list below are considered |
| 3615 | unsecure and shall not be used for any security-related function: |
| 3616 | a) SHA-1 |
| 3617 | b) MD5 |
| 3618 | c) RC4 |
| 3619 | d) RSA 1024 |
| 3620 | 4) Encrypted transmission between blocks or components – Even if critical Sensitive Data is |
| 3621 | stored in Secure Storage, any use of that data that requires its transmission out of that Secure |
| 3622 | Storage should be encrypted to prevent eavesdropping by malicious software within an |
| 3623 | MCU/MPU. |
| 3624 | 5) It is recommended to avoid using wildcard in Subject Id ("\*"), when setting up "/oic/sec/cred" |
| 3625 | Resource entries, since this opens up an identity spoofing opportunity. |
| 3626 | 6) Device vendor understands that it is the Device vendor’s responsibility to ensure the Device |
| 3627 | meets security requirements for its intended uses. As an example, IoTivity is a reference |
| 3628 | implementation intended to be used as a basis for a product, but IoTivity has not undergone |
| 3629 | 3rd party security review, penetration testing, etc. Any Device based on IoTivity should undergo |
| 3630 | appropriate penetration testing and security review prior to sale or deployment. |
| 3631 | 7) Device vendor agrees to publish the expected support lifetime for the Device to OCF and to |
| 3632 | consumers. Changes should be made to a public and accessible website. Expectations should |
| 3633 | be clear as to what will be supported and for how long the Device vendor expects to support |
| 3634 | security updates to the software, operating system, drivers, networking, firmware and hardware |
| 3635 | of the device. |
| 3636 | 8) Device vendor has not implemented test or debug interfaces on the Device which are operable |
| 3637 | or which can be enabled which might present an attack vector on the Device which circumvents |
| 3638 | the interface-level security or access policies of the Device. |
| 3639 | 9) Device vendor understands that if an application running on the Device has access to |
| 3640 | cryptographic elements such as the private keys or Ownership Credential, then those elements |
| 3641 | have become vulnerable. If the Device vendor is implementing a Bridge, an OBT, or a Device |
| 3642 | with access to the Internet beyond the local network, the execution of critical functions should |
| 3643 | take place within a Trusted or Secure Execution Environment (TEE/SEE). |
| 3644 | 10) Any PINs or fixed passphrases used for onboarding, Wi-Fi Easy Setup, SoftAP management or |
| 3645 | access, or other security-critical function, should be sufficiently unique (do not duplicate |
| 3646 | passphrases. The creation of these passphrases or PINS should not be algorithmically |
| 3647 | deterministic nor should they use insufficient entropy in their creation. |
| 3648 | 11) Ensure that there are no remaining "VENDOR\_TODO" items in the source code. |
| 3649 | 12) If the implementation of this document uses the "Just Works" onboarding method, understand |
| 3650 | that there is a man-in-the-middle vulnerability during the onboarding process where a malicious |
| 3651 | party could intercept messages between the device being onboarded and the OBT and could |
| 3652 | persist, acting as an intermediary with access to message traffic, during the lifetime of that |

|  |  |
| --- | --- |
| 3653 | onboarded device. The recommended best practice would be to use an alternate ownership |
| 3654 | transfer method (OTM) instead of "Just Works". |
| 3655 | 13) It is recommended that at least one static and dynamic analysis tool [1](#_bookmark294) be applied to any |
| 3656 | proposed major production release of the software before its release, and any vulnerabilities |
| 3657 | resolved. |
| 3658 | **14.2.3 Secure execution engine** |
| 3659 | Execution engine is the part of computing Platform that processes security functions, such as |
| 3660 | cryptographic algorithms or security protocols (e.g. DTLS). Securing the execution engine requires |
| 3661 | the following |
| 3662 | – Isolation of execution of sensitive processes from unauthorized parties/ processes. This |
| 3663 | includes isolation of CPU caches, and all of execution elements that needed to be considered |
| 3664 | as part of trusted (crypto) boundary. |
| 3665 | – Isolation of data paths into and out of execution engine. For instance, both unencrypted but |
| 3666 | sensitive data prior to encryption or after decryption, or cryptographic keys used for |
| 3667 | cryptographic algorithms, such as decryption or signing. See clause [14.2.4](#_bookmark291) for more details. |
| 3668 | **14.2.4 Trusted input/output paths** |
| 3669 | Platform implementations should only expose information, network interfaces, ports and other |
| 3670 | functions that are necessary for the correct functioning of the Platform. It is also strongly |
| 3671 | recommended that Vendors configure a Platform to expose only a fixed set of explicitly documented |
| 3672 | open network ports and/or port ranges. |
| 3673 | **14.2.5 Secure clock** |
| 3674 | Many security functions depend on time-sensitive credentials. Examples are time stamped |
| 3675 | Kerberos tickets, OAUTH tokens, X.509 certificates, OSCP response, software upgrades, etc. Lack |
| 3676 | of secure source of clock can mean an attacker can modify the system clock and fool the validation |
| 3677 | mechanism. Thus an SEE needs to provide a secure source of time that is protected from tampering. |
| 3678 | Trustworthiness from security robustness standpoint is not the same as accuracy. Protocols such |
| 3679 | as NTP can provide rather accurate time sources from the network, but are not immune to attacks. |
| 3680 | A secure time source on the other hand can be off by seconds or minutes depending on the time- |
| 3681 | sensitivity of the corresponding security mechanism. Secure time source can be external as long |
| 3682 | as it is signed by a trusted source and the signature validation in the local Device is a trusted |
| 3683 | process (e.g. backed by secure boot). |
| 3684 | **14.2.6 Approved algorithms** |
| 3685 | An important aspect of security of the entire ecosystem is the robustness of publicly vetted and |
| 3686 | peer-reviewed (e.g. NIST-approved) cryptographic algorithms. Security is not achieved by |
| 3687 | obscurity of the cryptographic algorithm. To ensure both interoperability and security, not only |
| 3688 | widely accepted cryptographic algorithms must be used, but also a list of approved cryptographic |
| 3689 | functions must be specified explicitly. As new algorithms are NIST approved or old algorithms are |
| 3690 | deprecated, the list of approved algorithms must be maintained by OCF. All other algorithms (even |
| 3691 | if they deemed stronger by some parties) must be considered non-approved. |
| 3692 | The set of algorithms to be considered for approval are algorithms for |
| 3693 | – Hash functions |
| 3694 | – Signature algorithms |
| 3695 | – Encryption algorithms |
| 3696 | – Key exchange algorithms |
| 3697 | – Pseudo Random functions (PRF) used for key derivation |

1 A general discussion of analysis tools can be found here: https[://w](http://www.ibm.com/developerworks/library/se-static/)ww[.ibm](http://www.ibm.com/developerworks/library/se-static/).[com/developerworks/library/se-static/](http://www.ibm.com/developerworks/library/se-static/)

|  |  |
| --- | --- |
| 3698 | This list will be included in this or a separate security robustness rules document and must be |
| 3699 | followed for all security specifications within OCF. |
| 3700 | **14.2.7 Hardware tamper protection** |
| 3701 | Various levels of hardware tamper protection exist. We borrow FIPS 140-2 terminology (not |
| 3702 | requirements) regarding tamper protection for cryptographic module |
| 3703 | – Production-grade (lowest level): this means components that include conformal sealing coating |
| 3704 | applied over the module’s circuitry to protect against environmental or other physical damage. |
| 3705 | This does not however require zeroization of secret material during physical maintenance. This |
| 3706 | definition is borrowed from FIPS 140-2 security level 1. |
| 3707 | – Tamper evident/proof (mid-level), This means the Device shows evidence (through covers, |
| 3708 | enclosures, or seals) of an attempted physical tampering. This definition is borrowed from FIPS |
| 3709 | 140-2 security level 2. |
| 3710 | – Tamper resistance (highest level), this means there is a response to physical tempering that |
| 3711 | typically includes zeroization of sensitive material on the module. This definition is borrowed |
| 3712 | from FIPS 140-2 security level 3. |
| 3713 | It is difficult of specify quantitative certification test cases for accreditation of these levels. Content |
| 3714 | protection regimes usually talk about different tools (widely available, specialized and professional |
| 3715 | tools) used to circumvent the hardware protections put in place by manufacturing. If needed, OCF |
| 3716 | can follow that model, if and when OCF engage in distributing sensitive key material (e.g. PKI) to |
| 3717 | its members. |
| 3718 | **14.3 Secure Boot** |
| 3719 | **14.3.1 Concept of software module authentication** |
| 3720 | In order to ensure that all components of a Device are operating properly and have not been |
| 3721 | tampered with, it is best to ensure that the Device is booted properly. There may be multiple stages |
| 3722 | of boot. The end result is an application running on top an operating system that takes advantage |
| 3723 | of memory, CPU and peripherals through drivers. |
| 3724 | The general concept is that each software module is invoked only after cryptographic integrity |
| 3725 | verification is complete. The integrity verification relies on the software module having been hashed |
| 3726 | (e.g. SHA\_1, SHA\_256) and then signed with a cryptographic signature algorithm with (e.g. RSA), |
| 3727 | with a key that only a signing authority has access to. |
| 3728 | [Figure 30](#_bookmark298) depicts software module authentication. |

Signer keys

Data

Hash function (e.g. SHA256)

Private key

Signature algorithm (RSA encryption, ECDSA)

Public Key Certificate

Data

Signature

Secure Storage/ TPM

|  |  |
| --- | --- |
| 3729 | **Figure 30 – Software Module Authentication** |
| 3730 | After the data is signed with the signer’s signing key (a private key), the verification key (the public |
| 3731 | key corresponding to the private signing key) is provided for later verification. For lower level |
| 3732 | software modules, such as bootloaders, the signatures and verification keys are inserted inside |
| 3733 | tamper proof memory, such as one-time programmable memory or TPM. For higher level software |
| 3734 | modules, such as application software, the signing is typically performed according to the PKCS#7 |
| 3735 | format [IETF RFC 2315,](#_bookmark9) where the signed data format includes both indications for signature |
| 3736 | algorithm, hash algorithm as well as the signature verification key (or certificate). Secure boot does |
| 3737 | not require use of PKCS#7 format. |
| 3738 | [Figure 31](#_bookmark299) depicts verification software module. |

Increasing Memory address

|  |
| --- |
| Data |
| Signature |
| Verification key |

|  |  |
| --- | --- |
| 3739 | **Figure 31 – Verification Software Module** |
| 3740 | As shown in [Figure 32](#_bookmark300) the verification module first decrypts the signature with the verification key |
| 3741 | (public key of the signer). The verification module also calculates a hash of the data and then |
| 3742 | compares the decrypted signature (the original) with the hash of data (actual) and if the two values |
| 3743 | match, the software module is authentic. |

Stored

Verification key

Stored signature

Data

Signature algorithm

(RSA decryption, ECDSA)

Hash function (e.g. SHA256)

Decrypted Signature Hashed Data

Match?

3744

3745

##### Figure 32 – Software Module Authenticity

**14.3.2 Secure Boot process**

|  |  |
| --- | --- |
| 3746 | Depending on the Device implementation, there may be several boot stages. Typically, in a PC/ |
| 3747 | Linux type environment, the first step is to find and run the BIOS code (first-stage bootloader) to |
| 3748 | find out where the boot code is and then run the boot code (second-stage boot loader). The second |
| 3749 | stage bootloader is typically the process that loads the operating system (Kernel) and transfers the |
| 3750 | execution to the where the Kernel code is. Once the Kernel starts, it may load external Kernel |
| 3751 | modules and drivers. |
| 3752 | When performing a secure boot, it is required that the integrity of each boot loader is verified before |
| 3753 | executing the boot loader stage. As mentioned, while the signature and verification key for the |
| 3754 | lowest level bootloader is typically stored in tamper-proof memory, the signature and verification |
| 3755 | key for higher levels should be embedded (but attached in an easily accessible manner) in the data |
| 3756 | structures software. |
| 3757 | **14.3.3 Robustness Requirements** |
| 3758 | **14.3.3.1 Robustness General** |
| 3759 | To qualify as high robustness secure boot process, the signature and hash algorithms shall be one |
| 3760 | of the approved algorithms, the signature values and the keys used for verification shall be stored |
| 3761 | in secure storage and the algorithms shall run inside a secure execution environment and the keys |
| 3762 | shall be provided the SEE over trusted path. |
| 3763 | **14.3.3.2 Next steps** |
| 3764 | Develop a list of approved algorithms and data formats |
| 3765 | **14.4 Attestation** |
| 3766 | **14.5 Software Update** |
| 3767 | **14.5.1 Overview** |
| 3768 | The Device lifecycle does not end at the point when a Device is shipped from the manufacturer; |
| 3769 | the distribution, retailing, purchase, installation/onboarding, regular operation, maintenance and |
| 3770 | end-of-life stages for the Device remain outstanding. It is possible for the Device to require update |
| 3771 | during any of these stages, although the most likely times are during onboarding, regular operation |

|  |  |
| --- | --- |
| 3772 | and maintenance. .The manufacturer shall have a defined policy available to OCF Security Domain |
| 3773 | Owner (e.g. via a website link) covering handling of any device vulnerabilities, including the |
| 3774 | software update information (e.g. if and how such updates are provided). This policy shall also |
| 3775 | cover any post end-of-life or end-of-service vulnerabilities. The aspects of the software include, but |
| 3776 | are not limited to, firmware, operating system, networking stack, application code, drivers, etc. |
| 3777 | **14.5.2 Recognition of Current Differences** |
| 3778 | Different manufacturers approach software update utilizing a collection of tools and strategies: |
| 3779 | over-the-air or wired USB connections, full or partial replacement of existing software, signed and |
| 3780 | verified code, attestation of the delivery package, verification of the source of the code, package |
| 3781 | structures for the software, etc. |
| 3782 | It is recommended that manufacturers review their processes and technologies for compliance with |
| 3783 | industry best-practices that a thorough security review of these takes place and that periodic review |
| 3784 | continue after the initial architecture has been established. |
| 3785 | This document applies to software updates as recommended to be implemented by OCF Devices; |
| 3786 | it does not have any bearing on the above-mentioned alternative proprietary software update |
| 3787 | mechanisms. The described steps are being triggered by an OCF Client, the actual implementation |
| 3788 | of the steps and how the software package is downloaded and upgraded is vendor specific. |
| 3789 | The triggers that can be invoked from OCF clients can: |
| 3790 | 1) Check if new software is available |
| 3791 | 2) Download and verify the integrity of the software package |
| 3792 | 3) Install the verified software package |
| 3793 | The triggers are not sequenced; each trigger can be invoked individually. |
| 3794 | The state of the transitions of software update is in [Figure 33.](#_bookmark307) |

3795

3796

3797

3798

3799

3800

3801

3802

##### Figure 33 – State transitioning diagram for software download

**Table 58 – Description of the software update bits**

|  |  |  |
| --- | --- | --- |
| **Bit** | **TM property** | **CM property** |
| Bit 9 | Initiate Software Availability Check | New Software Available |
| Bit 7 | Initiate Software Version Validation | Valid Software Available |
| Bit 8 | Initiate Secure Software Update | Upgrading |

##### 14.5.2.1 Checking availability of new software

Setting the Initiate Software Availability Check bit in the "/oic/sec/pstat.tm" Property (see [Table 38](#_bookmark253) of clause [13.8](#_bookmark251)) indicates a request to initiate the process to check if new software is available, e.g.

|  |  |
| --- | --- |
| 3803 | the process whereby the Device checks if a newer software version is available on the external |
| 3804 | endpoint. Once the Device has determined if an newer software version is available, it sets the |
| 3805 | Initiate Software Availability Check bit in the "/oic/sec/pstat.cm" Property to 1 (TRUE), indicating |
| 3806 | that new software is available or to 0 (FALSE) if no newer software version is available, See also |
| 3807 | [Table 58](#_bookmark308) where the bits in property TM indicates that the action is initiated and the CM bits are |
| 3808 | indicating the result of the action. The Device receiving this trigger is not downloading and not |
| 3809 | validating the software to determine if new software is available. The version check is determined |
| 3810 | by the current software version and the software version on the external endpoint. The |
| 3811 | determination if a software package is newer is vendor defined. |

3812

##### Software Version Validation

|  |  |
| --- | --- |
| 3813 | Setting the Initiate Software Version Validation bit in the "/oic/sec/pstat.tm" Property (see [Table 38](#_bookmark253) |
| 3814 | of [13.8](#_bookmark251)) indicates a request to initiate the software version validation process, the process whereby |
| 3815 | the Device validates the software (including firmware, operating system, Device drivers, networking |
| 3816 | stack, etc.) against a trusted source to see if, at the conclusion of the check, the software update |
| 3817 | process will need to be triggered (see clause [14.5.4](#_bookmark310)). When the Initiate Software Version Validation |
| 3818 | bit of "/oic/sec/pstat.tm" is set to 1 (TRUE) by a sufficiently privileged Client, the Device sets the |
| 3819 | "/oic/sec/pstat.cm" Initiate Software Version Validation bit to 0 and initiates a software version |
| 3820 | check. Once the Device has determined if a valid software is available, it sets the Initiate Software |
| 3821 | Version Validation bit in the "/oic/sec/pstat.cm" Property to 1 (TRUE) if an update is available or 0 |
| 3822 | (FALSE) if no update is available. To signal completion of the Software Version Validation process, |
| 3823 | the Device sets the Initiate Software Version Validation bit in the "/oic/sec/pstat.tm" Property back |
| 3824 | to 0 (FALSE). If the Initiate Software Version Validation bit of "/oic/sec/pstat.tm" is set to 0 (FALSE) |
| 3825 | by a Client, it has no effect on the validation process.The Software Version Validation process can |
| 3826 | download the software from the external endpoint to verify the integrity of the software package. |

3827

3828

3829

3830

3831

3832

3833

3834

3835

3836

3837

3838

3839

3840

3841

3842

3843

3844

3845

3846

3847

3848

3849

##### Software Update

The software of a Device shall be updatable.

Setting the Initiate Secure Software Update bit in the "/oic/sec/pstat.tm" Property (see [Table 38](#_bookmark253) of clause [13.8](#_bookmark251)) indicates a request to initiate the software update process. When the Initiate Secure Software Update bit of "/oic/sec/pstat.tm" is set to 1 (TRUE) by a sufficiently privileged Client, the Device sets the "/oic/sec/pstat.cm" Initiate Software Version Validation bit to 0 and initiates a software update process. Once the Device has completed the software update process, it sets the Initiate Secure Software Update bit in the "/oic/sec/pstat.cm" Property to 1 (TRUE) if/when the software was successfully updated or 0 (FALSE) if no update was performed. To signal completion of the Secure Software Update process, the Device sets the Initiate Secure Software Update bit in the "/oic/sec/pstat.tm" Property back to 0 (FALSE). If the Initiate Secure Software Update bit of "/oic/sec/pstat.tm" is set to 0 (FALSE) by a Client, it has no effect on the update process.

##### State of Device after software update

The state of all Resources implemented in the Device should be the same as after boot, meaning that the software update is not resetting user data and retaining a correct state.

User data of a Device is defined as:

* Retain the SVR states, e.g. the on boarded state, registered clients.
* Retain all created Resources
* Retain all stored data of a Resource

– For example the preferences stored for the brewing Resource ("oic.r.brewing").

##### Recommended Usage

The Initiate Secure Software Update bit of "/oic/sec/pstat.tm" should only be set by a Client after the Initiate Software Version Validation check is complete.

|  |  |
| --- | --- |
| 3850 | The process of updating Device software may involve state changes that affect the Device |
| 3851 | Operational State ("/oic/sec/pstat.dos"). Devices with an interest in the Device(s) being updated |
| 3852 | should monitor "/oic/sec/pstat.dos" and be prepared for pending software update(s) to affect Device |
| 3853 | state(s) prior to completion of the update. |
| 3854 | The Device itself may indicate that it is autonomously initiating a software version check/update or |
| 3855 | that a check/update is complete by setting the "pstat.tm" and "pstat.cm" Initiate Software Version |
| 3856 | Validation and Secure Software Update bits when starting or completing the version check or |
| 3857 | update process. As is the case with a Client-initiated update, Clients can be notified that an |
| 3858 | autonomous version check or software update is pending and/or complete by observing pstat |
| 3859 | Resource changes. |
| 3860 | The "oic.r.softwareupdate" Resource Type specifies additional features to control the software |
| 3861 | update process see core specification. |
| 3862 | **14.6 Non-OCF Endpoint interoperability** |
| 3863 | **14.7 Security Levels** |
| 3864 | Security Levels are a way to differentiate Devices based on their security criteria. This need for |
| 3865 | differentiation is based on the requirements from different verticals such as industrial and health |
| 3866 | care and may extend into smart home. This differentiation is distinct from Device classification |
| 3867 | (e.g. [IETF RFC 7228](#_bookmark22)) |
| 3868 | These categories of security differentiation may include, but is not limited to: |
| 3869 | 1) Security Hardening |
| 3870 | 2) Identity Attestation |
| 3871 | 3) Certificate/Trust |
| 3872 | 4) Onboarding Technique |
| 3873 | 5) Regulatory Compliance |
| 3874 | a) Data at rest |
| 3875 | b) Data in transit |
| 3876 | 6) Cipher Suites – Crypto Algorithms & Curves |
| 3877 | 7) Key Length |
| 3878 | 8) Secure Boot/Update |
| 3879 | In the future security levels can be used to define interoperability. |
| 3880 | The following applies to the OCF Security Specification 1.1: |
| 3881 | The current document does not define any other level beyond Security Level 0. All Devices will be |
| 3882 | designated as Level 0. Future versions may define additional levels. |
| 3883 | Additional comments: |
| 3884 | – The definition of a given security level will remain unchanged between versions of the document. |
| 3885 | – Devices that meet a given level may, or may not, be capable of upgrading to a higher level. |
| 3886 | – Devices may be evaluated and re-classified at a higher level if it meets the requirements of the |
| 3887 | higher level (e.g. if a Device is manufactured under the 1.1 version of the document, and a later |
| 3888 | document version defines a security level 1, the Device could be evaluated and classified as |
| 3889 | level 1 if it meets level 1 requirements). |
| 3890 | – The security levels may need to be visible to the End User. |

|  |  |
| --- | --- |
| 3891 | **14.8 Security Profiles** |
| 3892 | **14.8.1 Security Profiles General** |
| 3893 | Security Profiles are a way to differentiate OCF Devices based on their security criteria. This need |
| 3894 | for differentiation is based on the requirements from different verticals such as industrial and health |
| 3895 | care and may extend into smart home. This differentiation is distinct from device classification (e.g. |
| 3896 | [IETF RFC 7228](#_bookmark22)) |
| 3897 | These categories of security differentiation may include, but is not limited to: |
| 3898 | 1) Security Hardening and assurances criteria |
| 3899 | 2) Identity Attestation |
| 3900 | 3) Certificate/Trust |
| 3901 | 4) Onboarding Technique |
| 3902 | 5) Regulatory Compliance |
| 3903 | a) Data at rest |
| 3904 | b) Data in transit |
| 3905 | 6) Cipher Suites – Crypto Algorithms & Curves |
| 3906 | 7) Key Length |
| 3907 | 8) Secure Boot/Update |
| 3908 | Each Security Profile definition must specify the version or versions of the OCF Security |
| 3909 | Specification(s) that form a baseline set of normative requirements. The profile definition may |
| 3910 | include security requirements that supersede baseline requirements (not to relax security |
| 3911 | requirements). |
| 3912 | Security Profiles have the following properties: |
| 3913 | – A given profile definition is not specific to the version of the document that defines it. For |
| 3914 | example, the profile may remain constant for subsequent OCF Security Specification versions. |
| 3915 | – A specific OCF Device and platform combination may be used to satisfy the security profile. |
| 3916 | – Profiles may have overlapping criteria; hence it may be possible to satisfy multiple profiles |
| 3917 | simultaneously. |
| 3918 | – An OCF Device that satisfied a profile initially may be re-evaluated at a later time and found to |
| 3919 | satisfy a different profile (e.g. if a device is manufactured under the 1.1 version of the document, |
| 3920 | and a later document version defines a security profile Black, the device could be evaluated |
| 3921 | and classified as profile Black if it meets profile Black requirements). |
| 3922 | – A machine-readable representation of compliance results specifically describing profiles |
| 3923 | satisfied may be used to facilitate OCF Device onboarding. (e.g. a manufacturer certificate or |
| 3924 | manifest may contain security profiles attributes). |
| 3925 | **14.8.2 Identification of Security Profiles (Normative)** |
| 3926 | **14.8.2.1 Security Profiles in Prior Documents** |
| 3927 | OCF Devices conforming to versions of the OCF Security Specifications where Security Profiles |
| 3928 | Resource was not defined may be presumed to satisfy the "sp-baseline-v0" profile (defined in |
| 3929 | [14.8.3.3](#_bookmark320)) or may be regarded as unspecified. If Security Profile is unspecified, the Client may use |
| 3930 | the OCF Security Specification version to characterize expected security behaviour. |
| 3931 | **14.8.2.2 Security Profile Resource Definition** |
| 3932 | The "/oic/sec/sp" Resource is used by the OCF Device to show which OCF Security Profiles the |
| 3933 | OCF Device is capable of supporting and which are authorized for use by the OCF Security Domain |
| 3934 | owner. Properties of the Resource identify which OCF Security Profile is currently operational. The |

|  |  |
| --- | --- |
| 3935 | ocfSecurityProfileOID value type shall represent OID values and may reference an entry in the form |
| 3936 | of strings (UTF-8). |
| 3937 | "/oic/sec/sp" Resource is defined in [Table 59.](#_bookmark317) |
| 3938 | **Table 59 – Definition of the "/oic/sec/sp" Resource** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fixed URI** | **Resource Type Title** | **Resource Type ID ("rt" value)** | **OCF**  **Interfaces** | **Description** | **Related Functional Interaction** |
| /oic/sec/sp | Security Profile Resource Definition | oic.r.sp | oic.if.baselin e, oic.if.rw | Resource specifying supported and current security profile(s) | Discoverable |

3939

3940

3941

3942

3943

3944

3945

3946

3947

3948

3949

3950

3951

3952

3953

3954

3955

3956

3957

3958

3959

3960

3961

3962

3963

3964

3965

3966

3967

3968

3969

3970

3971

3972

3973

3974

[Table 60](#_bookmark318) defines the Properties of "/oic/sec/sp" Resource.

##### Table 60 – Properties of the "/oic/sec/sp" Resource

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Property Title** | **Property Name** | **Value Type** | **Value Rule** | **Access Mode** | **Mandatory** | **Description** |
| Supported Security Profiles | supportedprofil es | ocfSecur ityProfile OID | array | RW | Yes | Array of supported Security Profiles (e.g.  ["1.3.6.1.4.1.51414.0.0.2.0","1.3.6.1.4.1.514  14.0.0.3.0"]) |
| SecurityProfile | currentprofile | ocfSecur ityProfile OID | N/A | RW | Yes | Currently active Security Profile (e.g. "1.3.6.1.4.1.51414.0.0.3.0") |

The following OIDs are defined to uniquely identify Security Profiles. Future Security Profiles or changes to existing Security Profiles may result in a new ocfSecurityProfileOID.

id-OCF OBJECT IDENTIFIER ::= { iso(1) identified-organization(3) dod(6) internet(1)

private(4) enterprise(1) OCF(51414) } id-ocfSecurity OBJECT IDENTIFIER ::= { id-OCF 0 }

id-ocfSecurityProfile ::= { id-ocfSecurity 0 }

sp-unspecified ::= OBJECT IDENTIFIER { id-ocfSecurityProfile 0 }

--The Security Profile is not specified

sp-baseline ::= OBJECT IDENTIFIER { id-ocfSecurityProfile 1 }

--This specifies the OCF Baseline Security Profile(s)

sp-black ::= OBJECT IDENTIFIER { id-ocfSecurityProfile 2 }

--This specifies the OCF Black Security Profile(s)

sp-blue ::= OBJECT IDENTIFIER { id-ocfSecurityProfile 3 }

--This specified the OCF Blue Security Profile(s)

sp-purple ::= OBJECT IDENTIFIER { id-ocfSecurityProfile 4 }

--This specifies the OCF Purple Security Profile(s)

--versioned Security Profiles

sp-unspecified-v0 ::= ocfSecurityProfileOID (id-sp-unspecified 0}

--v0 of unspecified security profile, "1.3.6.1.4.1.51414.0.0.0.0" sp-baseline-v0 ::= ocfSecurityProfileOID {id-sp-baseline 0}

--v0 of baseline security profile, "1.3.6.1.4.1.51414.0.0.1.0" sp-black-v0 ::= ocfSecurityProfileOID {id-sp-black 0}

--v0 of black security profile, "1.3.6.1.4.1.51414.0.0.2.0" sp-blue-v0 ::= ocfSecurityProfileOID {id-sp-blue 0}

--v0 of blue security profile, "1.3.6.1.4.1.51414.0.0.3.0" sp-purple-v0 ::= ocfSecurityProfileOID {id-sp-purple 0}

--v0 of purple security profile, "1.3.6.1.4.1.51414.0.0.4.0" ocfSecurityProfileOID ::= UTF8String

|  |  |  |
| --- | --- | --- |
| 3975 | **14.8.3** | **Security Profiles** |
| 3976 | **14.8.3.1** | **Security Profiles General** |
| 3977 | The Security Profiles Resource shall be pre-populated with manufacturer default values (Refer to | |
| 3978 | the Security Profile clauses for additional details). | |
| 3979 | The OCF Conformance criteria may require vendor attestation that establishes the expected | |
| 3980 | environment in which the OCF Device is hosted (Refer to the Security Profile clauses for specific | |
| 3981 | requirements). | |
| 3982 | **14.8.3.2 Security Profile Unspecified (sp-unspecified-v0)** | |
| 3983 | The Security Profile "sp-unspecified-v0" is reserved for future use. | |
| 3984 | **14.8.3.3 Security Profile Baseline v0 (sp-baseline-v0)** | |
| 3985 | The Security Profile "sp-baseline-v0" is defined for all OCF Security Specification versions where | |
| 3986 | the "/oic/sec/sp" Resource is defined. All Devices shall include the "sp-baseline-v0" OID in the | |
| 3987 | "supportedprofiles" Property of the "/oic/sec/sp" Resource. | |
| 3988 | It indicates the OCF Device satisfies the normative security requirements for this document. | |
| 3989 | When a device supports the baseline profile, the "supportedprofiles" Property shall contain sp- | |
| 3990 | baseline-v0, represented by the OID string "1.3.6.1.4.1.51414.0.0.1.0", and may contain other | |
| 3991 | profiles. | |
| 3992 | When a manufacturer makes sp-baseline-v0 the default, by setting the "currentprofile" Property to | |
| 3993 | "1.3.6.1.4.1.51414.0.0.1.0", the "supportedprofiles" Property shall contain sp-baseline-v0. | |
| 3994 | **14.8.3.4 Security Profile Black (sp-black-v0)** | |
| 3995 | **14.8.3.4.1 Black Profile General** | |
| 3996 | The need for Security Profile Black v0 is to support devices and manufacturers who wish to certify | |
| 3997 | their devices meeting this specific set of security criteria. A Device may satisfy the Black | |
| 3998 | requirements as well as requirements of other profiles, the Black Security Profile is not necessarily | |
| 3999 | mutually exclusive with other Security Profiles unless those requirements conflict with the explicit | |
| 4000 | requirements of the Black Security Profile. | |
| 4001 | **14.8.3.4.2 Devices Targeted for Security Profile Black v0** | |
| 4002 | Security Profile Black devices could include any device a manufacturer wishes to certify at this | |
| 4003 | profile, but healthcare devices and industrial devices with additional security requirements are the | |
| 4004 | initial target. Additionally, manufacturers of devices at the edge of the network (or fog), or devices | |
| 4005 | with exceptional profiles of trust bestowed upon them, may wish to certify at this profile; these types | |
| 4006 | of devices may include, but are not limited to the following: | |
| 4007 | – Bridges (Mapping devices between ecosystems handling virtual devices from different | |
| 4008 | ecosystems) | |
| 4009 | – Resource Directories (Devices trusted to manage OCF Security Domain Resources) | |
| 4010 | – Remote Access (Devices which have external access but can also act within the OCF Security | |
| 4011 | Domain) | |
| 4012 | – Healthcare Devices (Devices with specific requirements for enhanced security and privacy) | |
| 4013 | – Industrial Devices (Devices with advanced management, security and attestation requirements) | |
| 4014 | **14.8.3.4.3 Requirements for Certification at Security Profile Black (Normative)** | |
| 4015 | Every device with "currentprofile" Property of the "/oic/sec/sp" Resource designating a Security | |
| 4016 | Profile of "sp-black-v0", as defined in clause [14.8.2,](#_bookmark316) must support each of the following: | |

|  |  |
| --- | --- |
| 4017 | – Onboarding via OCF Rooted Certificate Chain, including PKI chain validation |
| 4018 | – Support for AES 128 encryption for data at rest and in transit. |
| 4019 | – Hardening minimums: manufacturer assertion of secure credential storage |
| 4020 | – In [–](#_bookmark125) in enumerated item #10 “The "/oic/sec/cred" Resource should contain credential(s) if |
| 4021 | required by the selected OTM” is changed to require the credential be stored: “The |
| 4022 | "/oic/sec/cred" Resource shall contain credential(s).” |
| 4023 | – The OCF Device shall include an X.509v3 OCF Compliance Extension (clause [9.4.2.2.4](#_bookmark159)) in its |
| 4024 | certificate and the extension's 'securityProfile' field shall contain sp-black-v0 represented by |
| 4025 | the ocfSecurityProfileOID string, "1.3.6.1.4.1.51414.0.0.2.0". |
| 4026 | When a device supports the black profile, the "supportedprofiles" Property shall contain sp-black- |
| 4027 | v0, represented by the OID string "1.3.6.1.4.1.51414.0.0.2.0", and may contain other profiles. |
| 4028 | When a manufacturer makes sp-black-v0 the default, by setting the "currentprofile" Property to |
| 4029 | "1.3.6.1.4.1.51414.0.0.2.0", the "supportedprofiles" Property shall contain sp-black-v0. |
| 4030 | The OCF Rooted Certificate Chain and PKI Is defined by and structured within a framework |
| 4031 | described in the supporting documents: |
| 4032 | – Certificate Profile (See [9.4.2](#_bookmark151)) |
| 4033  4034 | – Certificate Policy (see Certificate Policy document: https://openconnectivity.org/specs/OCF%20Certificate%20Policy.pdf) |
| 4035 | **14.8.3.5 Security Profile Blue v0 (sp-blue-v0)** |
| 4036 | **14.8.3.5.1 Blue Profile General** |
| 4037 | The Security Profile Blue is used when manufacturers issue platform certificates for platforms |
| 4038 | containing manufacturer-embedded keys. Compatibility with interoperable trusted platforms is |
| 4039 | anticipated using certificate extensions defined by the Trusted Computing Group (TCG). OCF |
| 4040 | Security Domain owners evaluate manufacturer supplied certificates and attributed data to |
| 4041 | determine an appropriate OCF Security Profile that is configured for OCF Devices at onboarding. |
| 4042 | OCF Devices may satisfy multiple OCF Security Profiles. The OCF Security Domain owner may |
| 4043 | configure deployments using the Security Profile as OCF Security Domain partitioning criteria. |
| 4044 | Certificates issued to Blue Profile Devices shall be issued by a CA conforming to the CA Vetting |
| 4045 | Criteria defined by OCF. |
| 4046 | **14.8.3.5.2 Platforms and Devices for Security Profile Blue v0** |
| 4047 | The OCF Security Profile Blue anticipates an ecosystem where platform vendors may differ from |
| 4048 | OCF Device vendor and where platform vendors may implement trusted platforms that may conform |
| 4049 | to industry standards defining trusted platforms. The OCF Security Profile Blue specifies |
| 4050 | mechanisms for linking platforms with OCF Device(s) and for referencing quality assurance criteria |
| 4051 | produced by OCF conformance operations. The OCF Security Domain owner evaluates these data |
| 4052 | when an OCF Device is onboarded into the OCF Security Domain. Based on this evaluation the |
| 4053 | OCF Security Domain owner determines which Security Profile may be applied during OCF Device |
| 4054 | operation. All OCF Device types may be considered for evaluation using the OCF Security Profile |
| 4055 | Blue. |
| 4056 | **14.8.3.5.3 Requirements for Certification at Security Profile Blue v0** |
| 4057 | The OCF Device satisfies the Blue profile v0 (sp-blue-v0) when all of the security normative for this |
| 4058 | document version are satisfied and the following additional criteria are satisfied. |
| 4059 | OCF Blue profile defines the following OCF Device quality assurances: |

|  |  |
| --- | --- |
| 4060 | – The OCF Conformance criteria shall require vendor attestation that the conformant OCF Device |
| 4061 | was hosted on one or more platforms that satisfies OCF Blue platform security assurances and |
| 4062 | platform security and privacy functionality requirements. |
| 4063 | – The OCF Device achieving OCF Blue Security Profile compliance will be registered by OCF and |
| 4064 | published by OCF in a machine readable format. |
| 4065 | – The OCF Blue Security Profile compliance registry may be digitally signed by an OCF owned |
| 4066 | signing key. |
| 4067 | – The OCF Device shall include an X.509v3 OCF Compliance Extension (clause [9.4.2.2.4](#_bookmark159)) in its |
| 4068 | certificate and the extension's 'securityProfile' field shall contain sp-blue-v0 represented by the |
| 4069 | ocfSecurityProfileOID string, "1.3.6.1.4.1.51414.0.0.3.0". |
| 4070 | – The OCF Device shall include an X.509v3 OCF CPL Attributes Extension (clause [9.4.2.2.7](#_bookmark162)) in |
| 4071 | its certificate. |
| 4072 | – The DOTS is expected to perform a lookup of the certification status of the OCF Device using |
| 4073 | the OCF CPL Attributes Extension values and verify that the sp-blue-v0 OID is listed in the |
| 4074 | extension's "securityprofiles" field. |
| 4075 | OCF Blue profile defines the following OCF Device security functionality: |
| 4076 | – OCF Device(s) shall be hosted on a platform where a cryptographic and secure storage |
| 4077 | functions are hardened by the platform. |
| 4078 | – OCF Device(s) hosted on a platform shall expose accompanying manufacturer credentials using |
| 4079 | the "/oic/sec/cred" Resource where the "credusage" Property contains the value |
| 4080 | "oic.sec.cred.mfgcert". |
| 4081 | – OCF Device(s) that are hosted on a TCG-defined trusted platform should use an IEEE802.1AR |
| 4082 | IDevID and should verify the "TCG Endorsement Key Credential". All TCG-defined |
| 4083 | manufacturer credentials may be identified by the "oic.sec.cred.mfgcert" value of the |
| 4084 | "credusage" Property of the "/oic/sec/cred" Resource. They may be used in response to |
| 4085 | selection of the "oic.sec.doxm.mfgcert" owner transfer method. |
| 4086 | – OCF Device(s) shall use AES128 equivalent minimum protection for transmitted data. (See |
| 4087 | NIST SP 800-57). |
| 4088 | – OCF Device(s) shall use AES128 equivalent minimum protection for stored data. (See NIST SP |
| 4089 | 800-57). |
| 4090 | – OCF Device(s) should use AES256 equivalent minimum protection for stored data. (See NIST |
| 4091 | SP 800-57). |
| 4092 | – OCF Device(s) should protect the "/oic/sec/cred" Resource using the platform provided secure |
| 4093 | storage. |
| 4094 | – OCF Device(s) shall protect trust anchors (aka policy defining trusted CAs and pinned |
| 4095 | certificates) using platform provided secure storage. |
| 4096 | – OCF Device(s) should check certificate revocation status for locally issued certificates. |
| 4097 | – The DOTS is expected to check certificate revocation status for all certificates in manufacturer |
| 4098 | certificate path(s) if available. If a certificate is revoked, certificate validation fails and the |
| 4099 | connection is refused. The DOTS may disregard revocation status results if unavailable. |
| 4100 | OCF Blue profile defines the following platform security assurances: |
| 4101 | – Platforms implementing cryptographic service provider (CSP) functionality and secure storage |
| 4102 | functionality should be evaluated with a minimum FIPS140-2 Level 2 or Common Criteria EAL |
| 4103 | Level 2. |
| 4104 | – Platforms implementing trusted platform functionality should be evaluated with a minimum |
| 4105 | Common Criteria EAL Level 1. |
| 4106 | OCF Blue profile defines the following platform security and privacy functionality: |

|  |  |
| --- | --- |
| 4107 | – The Platform shall implement cryptographic service provider (CSP) functionality. |
| 4108 | – Platform CSP functionality shall include cryptographic algorithms, random number generation, |
| 4109 | secure time. |
| 4110 | – The Platform shall implement AES128 equivalent protection for transmitted data. (See NIST SP |
| 4111 | 800-57). |
| 4112 | – The Platform shall implement AES128 and AES256 equivalent protection for stored data. (See |
| 4113 | NIST SP 800-57). |
| 4114 | – Platforms hosting OCF Device(s) should implement a platform identifier following IEEE802.1AR |
| 4115 | or Trusted Computing Group(TCG) specifications. |
| 4116 | – Platforms based on Trusted Computing Group (TCG) platform definition that host OCF Device(s) |
| 4117 | should supply TCG-defined manufacture certificates; also known as "TCG Endorsement Key |
| 4118 | Credential" (which complies with [IETF RFC 5280](#_bookmark16)) and "TCG Platform Credential" (which |
| 4119 | complies with [IETF RFC 5755](#_bookmark19)). |
| 4120 | When a device supports the blue profile, the "supportedprofiles" Property shall contain sp-blue-v0, |
| 4121 | represented by the OID string "1.3.6.1.4.1.51414.0.0.3.0", and may contain other profiles. |
| 4122 | When a manufacturer makes sp-blue-v0 the default, by setting the "currentprofile" Property to |
| 4123 | "1.3.6.1.4.1.51414.0.0.3.0", the "supportedprofiles" Property shall contain sp-blue-v0. |
| 4124 | During onboarding, while the device state is RFOTM, the DOTS may update the "currentprofile" |
| 4125 | Property to one of the other values found in the "supportedprofiles" Property. |
| 4126 | **14.8.3.6 Security Profile Purple v0 (sp-purple-v0)** |
| 4127 | Every device with the "/oic/sec/sp" Resource designating "sp-purple-v0", as defined in clause |
| 4128 | [14.8.2](#_bookmark316) must support following minimum requirements |
| 4129 | – Hardening minimums: secure credential storage, software integrity validation, secure update. |
| 4130 | – If a Certificate is used, the OCF Device shall include an X.509v3 OCF Compliance Extension |
| 4131 | (clause [9.4.2.2.4](#_bookmark159)) in its certificate and the extension's 'securityProfile' field shall contain sp- |
| 4132 | purple-v0 represented by the ocfSecurityProfileOID string, "1.3.6.1.4.1.51414.0.0.4.0" |
| 4133 | – The OCF Device shall include a X.509v3 OCFCPLAttributes Extension (clause [9.4.2.2.7](#_bookmark162)) in its |
| 4134 | End-Entity Certificate when manufacturer certificate is used. |
| 4135 | Security Profile Purple has following optional security hardening requirements that the device can |
| 4136 | additionally support. |
| 4137 | – Hardening additions: secure boot, hardware backed secure storage |
| 4138 | – The OCF Device shall include a X.509v3 OCFSecurityClaims Extension (clause [9.4.2.2.6](#_bookmark161)) in its |
| 4139 | End-Entity Certificate and it shall include corresponding OIDs to the hardening additions |
| 4140 | implemented and attested by the vendor. If there is no additional support for hardening |
| 4141 | requirements, X.509v3 OCFSecurityClaims Extension shall be omitted. |
| 4142 | For software integrity validation, OCF Device(s) shall provide the integrity validation mechanism |
| 4143 | for security critical executables such as cryptographic modules or secure service applications, and |
| 4144 | they should be validated before the execution. The key used for validating the integrity must be |
| 4145 | pinned at the least to the validating software module. |
| 4146 | For secure update, OCF Device(s) shall be able to update its firmware in a secure manner. |
| 4147 | For secure boot, OCF Device(s) shall implement the BIOS code (first-stage bootloader on ROM) to |
| 4148 | be executed by the processor on power-on, and secure boot parameters to be provisioned by |
| 4149 | tamper-proof memory. Also OCF Device(s) shall provide software module authentication for the |
| 4150 | security critical executables and stop the boot process if any integrity of them is compromised. |

|  |  |
| --- | --- |
| 4151 | For hardware backed secure storage, OCF Device(s) shall store sensitive data in non-volatile |
| 4152 | memory ("NVRAM") and prevent the retrieval of sensitive data through physical and/or electronic |
| 4153 | attacks. |
| 4154 | More details on security hardening guidelines for software integrity validation, secure boot, secure |
| 4155 | update, and hardware backed secure storage are described in [14.3,](#_bookmark296) [14.5](#_bookmark304) and [14.2.2.2.](#_bookmark289) |
| 4156 | Certificates issued to Purple Profile Devices shall be issued by a CA conforming to the CA Vetting |
| 4157 | Criteria defined by OCF. |
| 4158 | When a device supports the purple profile, the "supportedprofiles" Property shall contain sp-purple- |
| 4159 | v0, represented by the OID string "1.3.6.1.4.1.51414.0.0.4.0", and may contain other profiles. |
| 4160 | When a manufacturer makes sp-purple-v0 the default, by setting the "currentprofile" Property to |
| 4161 | "1.3.6.1.4.1.51414.0.0.4.0", the "supportedprofiles" Property shall contain sp-purple-v0. |

|  |  |
| --- | --- |
| 4162 | **15 Device Type Specific Requirements** |
| 4163 | **15.1 Bridging Security** |
| 4164 | **15.1.1 Universal Requirements for Bridging to another Ecosystem** |
| 4165 | The Bridge shall go through OCF ownership transfer as any other onboardee would. |
| 4166 | The software of a Bridge shall be field updatable. (This requirement need not be tested but can be |
| 4167 | certified via a vendor declaration.) |
| 4168 | Each VOD shall be onboarded by an OCF OBT. Each Virtual Bridged Device should be provisioned |
| 4169 | as appropriate in the Bridged Protocol. In other words, VODs and Virtual Bridged Devices are |
| 4170 | treated the same way as physical Devices. They are entities that have to be provisioned in their |
| 4171 | network. |
| 4172 | Each VOD shall implement the behaviour required by [ISO/IEC 30118-1](#_bookmark3) and this document. Each |
| 4173 | VOD shall perform authentication, access control, and encryption according to the security settings |
| 4174 | it received from the OCF OBT. Each Virtual Bridged Device shall implement the security |
| 4175 | requirements of the Bridged Protocol. |
| 4176 | In addition, in order to be considered secure from an OCF perspective, the Bridge Platform shall |
| 4177 | use appropriate ecosystem-specific security options for communication between the Virtual Bridged |
| 4178 | Devices instantiated by the Bridge and Bridged Devices. This security shall include mutual |
| 4179 | authentication, and encryption and integrity protection of messages in the bridged ecosystem. |
| 4180 | A VOD may authenticate itself to the DOTS using the Manufacturer Certificate Based OTM (see |
| 4181 | clause [7.3.6](#_bookmark107)) with the Manufacturer Certificate and corresponding private key of the Bridge which |
| 4182 | instantiated that VOD. |
| 4183 | A VOD may authenticate itself to the OCF Cloud using the Manufacturer Certificate and |
| 4184 | corresponding private key of the Bridge which instantiated that VOD. |
| 4185 | A Bridge and the VODs created by that Bridge shall operate as independent Devices, with the |
| 4186 | following exceptions: |
| 4187 | – If a Bridge creates a VOD while the Bridge is in an Unowned State, then the VOD shall be |
| 4188 | created in an Unowned State. |
| 4189 | – An Unowned VOD shall not accept DTLS connection attempts nor TLS connection attempts nor |
| 4190 | any other requests, including discovery requests, while the Bridge (that created that VOD) is |
| 4191 | Unowned. |
| 4192 | – At any time when a Bridge is transitioning from Owned to Unowned State, all Unowned VODs |
| 4193 | (created by that Bridge prior to the transition) shall drop any existing TLS and/or DTLS |
| 4194 | connections. |
| 4195 | – At any time when a Bridge is transitioning from Unowned to Owned State, the Bridge shall |
| 4196 | trigger all Unowned VODs (created by that Bridge prior to the transition) to become accessible |
| 4197 | in RFOTM, with internal state as if the VOD has just transitioned from RESET to RFOTM. |
| 4198 | – If a Bridge creates a VOD while the Bridge is in an Owned State, then the VOD shall become |
| 4199 | accessible in RFOTM, with internal state as if the VOD has just transitioned from RESET to |
| 4200 | RFOTM. |
| 4201 | [Table 61](#_bookmark324) intends to clarify this behaviour. |

4202

4203

##### Table 61 – Dependencies of VOD Behaviour on Bridge state, as clarification of accompanying text

|  |  |  |
| --- | --- | --- |
| **Bridge state** | **Additional dependencies on VOD behaviour** | |
| **VOD is Unowned (either just created, or created previously)** | **VOD is Owned** |
| From unboxing Bridge until just prior to the end of transition of Bridge from Unowned to Owned | No accepting DTLS connection attempts nor TLS connection attempts nor any other requests, including discovery requests | Not applicable |
| At end of transition from Unowned to Owned | VOD becomes accessible in RFOTM following Bridge's transition. Internal state as if just  transitioned from RESET. | As per normal Device |
| Owned | As per normal Device | As per normal Device |
| At Start of transition from Owned to Unowned | Drop any established TLS/DTLS connections,  even if already partway through Device ownership | As per normal Device |
| Start of transition from Owned to Unowned, until just prior to the end of transition from Unowned to Owned. | No accepting DTLS connection attempts nor TLS connection attempts nor any other requests, including discovery requests | As per normal Device |

|  |  |
| --- | --- |
| 4204 | The "vods" Property of the "oic.r.vodlist" Resource on a Bridge reflects the details of all currently |
| 4205 | Owned VODs which have been created by that Bridge since the most recent hardware reset (if any) |
| 4206 | of the Bridge Platform (which removes all the created VODs), regardless of whether the VODs have |
| 4207 | the same owner as the Bridge or not. The entries in the "vods" Property are added and removed |
| 4208 | according to the following criteria: |
| 4209 | – Whenever a VOD created by a Bridge transitions from being Unowned to being Owned, then |
| 4210 | an entry for that VOD shall be added to the "vods" Property of the "oic.r.vodlist" Resource of |
| 4211 | that Bridge. |
| 4212 | – Whenever a VOD created by a Bridge transitions from being Owned to being Unowned, then |
| 4213 | entry for that VOD shall be removed from the "vods" Property of the "oic.r.vodlist" Resource of |
| 4214 | that Bridge. If that Bridge is currently in Unowned state, then the "oic.r.vodlist" Resource is not |
| 4215 | accessible, and the entry for that VOD shall be removed from the "vods" Property before or |
| 4216 | during the transition of that Bridge to the Owned state. |
| 4217 | – All other modifications of the list are not allowed. |
| 4218 | A Bridge shall only expose a secure OCF Endpoint for the "oic.r.vodlist" Resource. |
| 4219 | **15.1.2 Additional Security Requirements specific to Bridged Protocols** |
| 4220 | **15.1.2.1 Additional Security Requirements specific to the AllJoyn Protocol** |
| 4221 | For AllJoyn translator, an authenticated and authorized Client shall be able to block the |
| 4222 | communication of all OCF Devices with all Bridged Devices that don’t communicate securely with |
| 4223 | the Bridge, by using the Bridge Device’s "oic.r.securemode" Resource specified in [ISO/IEC 30118-](#_bookmark4) |
| 4224 | [3](#_bookmark4) |
| 4225 | **15.1.2.2 Additional Security Requirements specific to the Bluetooth LE Protocol** |
| 4226 | A Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t |
| 4227 | communicate securely with the Bridge. |
| 4228 | **15.1.2.3 Additional Security Requirements specific to the oneM2M Protocols** |
| 4229 | The Bridge shall implement oneM2M application access control as defined in the [oneM2M Release](#_bookmark28) |
| 4230 | [3 Specifications.](#_bookmark28) |
| 4231 | An Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t |
| 4232 | communicate securely with the Bridge. |

4233

4234

4235

4236

4237

4238

4239

4240

4241

4242

4243

4244

4245

4246

4247

4248

4249

4250

4251

4252

4253

4254

4255

4256

4257

4258

4259

4260

4261

4262

4263

4264

4265

##### Additional Security Requirements specific to the U+ Protocol

A Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t communicate securely with the Bridge.

##### Additional Security Requirements specific to the Z-Wave Protocol

A Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t communicate securely with the Bridge.

##### Additional Security Requirements specific to the Zigbee Protocol

A Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t communicate securely with the Bridge.

##### Additional Security Requirements specific to the EnOcean Radio Protocol

A Bridge shall block the communication of all OCF Devices with all Bridged Devices that don’t communicate securely with the Bridge.

.

|  |  |
| --- | --- |
| 4266 | **Annex A** |
| 4267 | **(informative)** |
| 4268 | **Access Control Examples** |
| 4269 | **16 Alternative in-transit protection mechanisms** |
| 4270 | **16.1 Introduction to in-transit protection mechanisms** |
| 4271 | In addition to the DTLS protection mechanisms for device-to-device communication specified in |
| 4272 | clause [10](#_bookmark169) and clause [11.2,](#_bookmark182) and TLS protection specified in [OCF Cloud Security Specification,](#_bookmark7) OCF |
| 4273 | supports the following in-transit protection mechanisms: |
| 4274 | – End-to-End Security of Unicast Messages using OSCORE, specified in clause [16.2.](#_bookmark330) |
| 4275 | – Simple Secure Multicast, specified in clause [16.3](#_bookmark336) |
| 4276 | **16.2 End-to-End Security of Unicast Messages using OSCORE** |
| 4277 | **16.2.1 Introduction to End-to-End Security of Unicast Messages using OSCORE** |
| 4278 | End-to-End Security of Unicast Messages is accomplished by applying a layer of in-transit |
| 4279 | protection above the transport layer Security (provided by DTLS or TLS) and below the resource- |
| 4280 | access authorization layer, using Object Security for Constrained RESTful Environments (OSCORE) |
| 4281 | [IETF RFC 8613.](#_bookmark27) |
| 4282 | Relative to an exchange of an OCF CRUDN Request message and OCF CRUDN Response |
| 4283 | message: |
| 4284 | – The Device acting as a Client (that is, sending an OCF CRUDN Request message and receiving |
| 4285 | the corresponding OCF CRUDN Response message) acts as an OSCORE client. Within the |
| 4286 | scope of clause [16.2,](#_bookmark330) all Clients are assumed to support OSCORE and perform OSCORE client |
| 4287 | processing. |
| 4288 | – The Device acting as a Server (that is, receiving an OCF CRUDN Request message and |
| 4289 | sending one or more corresponding OCF CRUDN Response messages) acts as an OSCORE |
| 4290 | server. Within the scope of clause [16.2,](#_bookmark330) all Servers are assumed to support OSCORE and |
| 4291 | perform OSCORE server processing. |
| 4292 | Clause [16.2.4](#_bookmark335) specifies the supported mechanism for establishing an OSCORE Security Context |
| 4293 | between two Devices. For each Device, an authorized Client (e.g. OBT) provisions the OSCORE |
| 4294 | Security Context parameters to a credential entry of the "/oic/sec/cred" Resource. The |
| 4295 | "subjectuuid" of that credential entry identifies the other Device that shares that OSCORE Security |
| 4296 | Context (similar to how a DTLS endpoint associates each DTLS PSK session with the Device UUID |
| 4297 | of the other DTLS endpoint). |
| 4298 | **16.2.2 OSCORE ID Namespace Prefix** |
| 4299 | Clause [16.2.4](#_bookmark335) specifies one mechanism for establishing an OSCORE Security Context between |
| 4300 | two Devices. Different mechanisms have different entities responsible for managing the selection |
| 4301 | of OSCORE Sender ID and OSCORE Recipient ID. There is value in preventing Devices having |
| 4302 | multiple OSCORE Security Contexts with identical Recipient IDs: this simplifies processing and |
| 4303 | avoids inefficiencies. |
| 4304 | If a set of one or more coordinated entities (e.g. a group of OBTs) assigns a set of OSCORE |
| 4305 | Recipient IDs to OSCORE Security Contexts on a Device, then that set of entities is responsible |
| 4306 | for avoiding duplicate OSCORE Recipient IDs. However, two non-coordinated entities assigning |
| 4307 | OSCORE Recipient IDs might assign identical OSCORE Recipient IDs if there is no predefined |
| 4308 | agreement on assignment of OSCORE Recipient IDs. |
| 4309 | For this reason, the first byte of the OSCORE Sender ID and OSCORE Recipient ID use a OSCORE |
| 4310 | Identifier Namespace Prefix. The Table Y is the authoritative definition of the assigned OSCORE |
| 4311 | Identifier Namespace Prefix values. |

4312

4313

##### Table 62 – OSCORE Identifier Namespace Prefix

|  |  |  |
| --- | --- | --- |
| **Value** | **Interpretation** | **Applicable clauses** |
| 0x00 | Reserved for future use |  |
| 0x01 | Directly provisioned OSCORE Security Context | [16.2.4](#_bookmark335) |
| 0x02 | Simple Secure Multicast | [16.3](#_bookmark336) |
| 0x03-0x0F | Reserved for future use |  |

**16.2.3 OSCORE protection and verification of unicast OCF CRUDN messages**

|  |  |
| --- | --- |
| 4314 | All OSCORE message processing requirements in clause 8 in [IETF RFC 8613](#_bookmark27) apply. |
| 4315 | NOTE 1: Clause 8 in [IETF RFC 8613](#_bookmark27) requires the Client keep the association of the request Token (see [IETF RFC 7252)](#_bookmark24) |
| 4316 | with the Security Context and Partial IV of the request, in order to be able to find the Security Context and compute the |
| 4317 | OSCORE Additional Authenticated Data when verifying the response. |
| 4318 | If a Client has an established OSCORE Security Context associated with a Server, then the |
| 4319 | following call flow applies whenever the Client sends unicast OCF CRUDN request targeting |
| 4320 | Resources hosted on the Server. The Client may send multiple OSCORE requests to multiple |
| 4321 | Servers |
| 4322 | 1) The Client shall apply the OSCORE request protection processing to OCF CRUDN requests |
| 4323 | targeting Resources hosted on the Server as specified in clause 8.1 in [IETF RFC 8613,](#_bookmark27) using |
| 4324 | the OSCORE Security Context. See [ISO/IEC 30118-1](#_bookmark3) for details on setting the Proxy-URI |
| 4325 | option. |
| 4326 | The Client sends the OSCORE request message to the Server (optionally via OCF Proxies). |
| 4327 | The OSCORE request message shall be delivered over secure transports: Device-to-Device |
| 4328 | communication is secured as specified in clause [10;](#_bookmark169) Device to Cloud communication is secured |
| 4329 | as specified in [OCF Cloud Specification](#_bookmark6) and [OCF Cloud Security Specification;](#_bookmark7) and Cloud-to- |
| 4330 | Cloud communication is secured as specified in OCF Cloud API for Cloud Services |
| 4331 | Specification. |
| 4332 | 2) The Server receives a unicast OSCORE request message. The Server shall apply the OSCORE |
| 4333 | request verification and decryption processing in clause 8.2 of [IETF RFC 8613](#_bookmark27) with the |
| 4334 | following clarifications: |
| 4335 | a) At Step 2 in clause 8.2 of [IETF RFC 8613](#_bookmark27) |
| 4336 | i) If either the decompression or the COSE message fails to decode, the Server shall |
| 4337 | respond with error response message (e.g. "Bad Option") including an Outer Max-Age |
| 4338 | option with value zero. |
| 4339 | ii) The Server attempts to retrieve the OSCORE Security Contexts associated with the |
| 4340 | Recipient ID in the ’kid’ parameter. If the Server fails to retrieve a OSCORE Security |
| 4341 | Context with OSCORE Recipient ID corresponding to the 'kid' parameter received, then |
| 4342 | the Server shall respond with an error response message (e.g. "Unauthorized") |
| 4343 | including an Outer Max-Age option with value zero. |
| 4344 | b) At step 6 in clause 8.2 of [IETF RFC 8613,](#_bookmark27) if the decryption failed then the Server shall |
| 4345 | respond with an error response message (e.g. "Bad Request) including an Outer Max-Age |
| 4346 | option with value zero. |
| 4347 | c) If a Server exposes one or more observable Resources, then the Server shall support |
| 4348 | receiving OSCORE request messages using the Observe option. |
| 4349 | 3) The Server shall process the OCF CRUDN request message (encapsulated in the OSCORE |
| 4350 | request message) resulting in OCF CRUDN response message(s). The Server shall treat the |
| 4351 | value of "subjectuuid" in the credential entry which contains the OSCORE Security Context |
| 4352 | used to verify and decrypt the OSCORE request message in Step 2 as Client's Device UUID |
| 4353 | for access control processing. The Server shall treat the connection type as "auth-crypt" for |
| 4354 | access control processing. |

|  |
| --- |
| NOTE 2: Multiple OCF CRUDN response messages are only sent in scenarios where the OCF CRUDN Request message |
| is an Observe Request message. |
| 4) The Server shall apply the OSCORE response protection processing of clause 8.3 of [IETF RFC](#_bookmark27) |
| [8613](#_bookmark27) to each OCF CRUDN response message, using the OSCORE Security Context used to |
| successfully decrypt the OSCORE request (in Step 2 of the present clause). |
| At Step 3 in clause 8.3 of [IETF RFC 8613,](#_bookmark27) the Server shall compute the AEAD nonce as |
| described in clause 5.2 of [IETF RFC 8613](#_bookmark27) by applying the following steps: |
| a) Encode the Partial IV (OSCORE Sender Sequence Number in network byte order) and |
| increment the OSCORE Sender Sequence Number by one. |
| b) Compute the OSCORE AEAD nonce from the Sender ID, Common IV, and Partial IV. |
| The Server shall support sending the OCF CRUDN response messages using the Observe |
| option in OSCORE response messages. If an OCF CRUDN response message uses the |
| Observe option, then the OSCORE response message shall include an Outer Max-Age option |
| with value zero. The Server sends the OSCORE response message to the Client (optionally via |
| OCF Proxies). As with the OSCORE request message, the OSCORE response message shall |
| be delivered over secure transports - see Step 1 for details. |
| The Server shall update the value of the "ssn" Property in the matching credential entry of the |
| "/oic/sec/cred" Resource to reflect the next value of the OSCORE Sender Sequence Number |
| to be sent to a corresponding Endpoint. |
| NOTE 3: If a Client retrieves the "/oic/sec/cred" Resource over the OSCORE channel, the OSCORE Sender Sequence |
| Number in the header of the OSCORE message is expected to match the "ssn" value within the Resource representation. |
| 5) The Client receives the OSCORE response message. The Client uses the Token (see [IETF](#_bookmark24) |
| [RFC 7252](#_bookmark24)) in this response message to determine the corresponding OCF CRUDN request |
| message, the OSCORE Security Context and Partial IV in Step 1 of the present clause; see |
| Note 1. The Client shall apply OSCORE response protection processing of clause 8.3 of [IETF](#_bookmark27) |
| [RFC 8613](#_bookmark27) using this OSCORE Security Context and Partial IV. The Client should ignore a |
| success response to an OSCORE-protected request if the response is not an OSCORE |
| response message (indicated by the presence of the OSCORE option). |
| **16.2.4 Direct provisioning of an OSCORE Security Context** |
| This is a mechanism for establishing an OSCORE Security Context for communication between |
| two Endpoints. All configurable parameters of the OSCORE Security Context are either: |
| – fixed to the OSCORE-specified default value, or |
| – directly provisioned by an authorized Client (e.g. OBT) to a credential entry of the |
| "/oic/sec/cred" Resource of the two Endpoints. |
| The following OSCORE Security Context parameters shall use the default values defined in clause |
| 3.2 of [IETF RFC 8613](#_bookmark27) (this information is not configured by the OBT): |
| – AEAD Algorithm, |
| – HKDF, |
| – Replay Window, |
| – Master Salt, |
| – ID Context. |
| The following OSCORE Security Context parameters and associated Device UUID shall be |
| provisioned to a credential entry of "/oic/sec/cred" of the Device: |
| – The "subjectuuid" shall be set to the deviceUUID of the other Endpoint to be associated with |
| the OSCORE Security Context. |
| – The "credtype" shall be set to the value specified for a directly provisioned OSCORE Security |
| Context in [Table 21,](#_bookmark224) clause [13.3.1.](#_bookmark221) |

|  |
| --- |
| – The "privatedata" Property of the credential entry shall be set to the 256-bit secret generated |
| by the provisioning client (e.g. OBT). This value shall be used as the OSCORE Master Secret. |
| Two Endpoints provisioned using this mechanism can communicate securely only if provisioned |
| with identical values for the OSCORE Master Secret. |
| – The OSCORE Configuration parameters ("oscore") Property shall be present, and shall include |
| the following Properties: |
| – The OSCORE Sender ID of the OSCORE Security Context is in the "senderid" Property. |
| That value shall be set to the hexadecimal representation of a 56-bit value selected by the |
| provisioning Client (e.g. OBT). When using the mechanism described in the present clause, |
| the first byte of this value is expected to have the value assigned in [Table 62](#_bookmark333) for a directly |
| provisioned OSCORE Security Context. |
| – The OSCORE Recipient ID of the OSCORE Security Context is in the "recipientid" Property. |
| That value shall be set to the hexadecimal representation of a 56-bit value selected by the |
| provisioning Client (e.g. OBT). The first byte of this value is expected to have the value |
| assigned in [Table 62](#_bookmark333) for a directly provisioned OSCORE Security Context. |
| NOTE: The values for the OSCORE Sender ID and OSCORE Recipient ID of the OSCORE Security Context for one |
| Device are provisioned as the values for the OSCORE Recipient ID and OSCORE Sender ID of the OSCORE Security |
| Context for the other Device respectively. |
| On Device powering down, for each such credential entry, the Device shall write the value of |
| corresponding OSCORE Sender Sequence Number as "ssn" Property to non-volatile memory. In |
| event of a crash, devices should apply Appendix B.1.1 of [IETF RFC 8613.](#_bookmark27) |
| **16.3 Simple Secure Multicast** |
| **16.3.1 Introduction to Simple Secure Multicast** |
| The communication model is that one (1) Client communicates to a group of Servers with a single |
| UPDATE request, as shown in [Figure 34.](#_bookmark338) Each Server receives the UPDATE request at |
| approximately the same time and can execute the UPDATE request at approximately the same |
| time. As example of this kind of communication is sending an “on” command to a group of lights, |
| all lights that are member of that group turn on at approximately the same time. Sending UPDATE |

4430

4431

requests to a group of devices can be achieved on IP by means of sending messages to a predefined URL on a multicast address.



Configure

Servers

Configure

1 Client

Reaching a group of Ser

Multicast command

Client

(single client in the deployment)

vers

Multi cast Group

OBT

Server

Server

4432

4433

##### Figure 34 – Simple Multicast requests

|  |
| --- |
| Security of SSM is accomplished by applying an application layer of in-transit protection and below |
| the resource-access authorization layer, using OSCORE [IETF RFC 8613.](#_bookmark27) |
| Relative to an exchange of an UPDATE non-confirmable message: |
| – The Device acting as a Client (that is, sending an UPDATE request message) acts as an |
| OSCORE client. Within the scope of clause [16.3](#_bookmark336) the single Client is assumed to support |
| OSCORE and perform OSCORE client processing. |
| – The Device acting as a Server (that is, receiving an UPDATE request message) acts as an |
| OSCORE server. Within the scope of clause [16.3,](#_bookmark336) all Servers are assumed to support OSCORE |
| and perform OSCORE server processing. |
| Clause [16.3.2](#_bookmark339) details the assumptions and prerequisites for correct functioning of SSM. Clause |
| [16.3.3](#_bookmark340) describes the process for encapsulating an UPDATE request message into an SSM Request |
| at the Client of an SSM Group, and subsequent extraction of an UPDATE request message from |
| an SSM Request at the Server of an SSM Group. Clause [16.3.4](#_bookmark341) specifies how a Client generates |
| an OSCORE Common Context and OSCORE Sender Context from an SSM Client Context and |
| specifies how a Server generates an OSCORE Common Context and OSCORE Recipient Context |
| from an SSM Server Context. |
| **16.3.2 Assumptions and Prerequisites for Simple Secure Multicast** |
| As shown in the following example, any Server of the SSM Group can generate an SSM Request |
| which other Servers in the SSM Group will interpret as being securely sent by the Client of the |
| SSM Group, for the purposes of privilege escalation. The security of SSM relies on the assumption |
| that no Server in the SSM Group attempts to generate an SSM Request using the credentials for |
| the SSM Group. SSM should only be used in scenarios where the Security Domain Owner is |
| confident that this is a valid assumption. |
| SSM Requests are delivered to SSM-capable Servers via the All OCF Nodes multicast address |
| defined in [ISO/IEC 30118-1.](#_bookmark3) As specified in [ISO/IEC 30118-1,](#_bookmark3) all Servers subscribe to this multicast |
| address to facilitate discovery of "oic/res", and consequently all Servers can receive SSM Requests |
| delivered in this manner. A Server that supports the reception of SSM Requests for one or more |
| Resources that it hosts shall populate the All OCF nodes multicast address in the "eps" Parameter |
| of the Resource Links of those Resources in the "oic/res" discovery response. |
| The configured Client is aware of Multicast enabled Servers by means of detecting the multicast |
| enabled resources in the Device discovery "oic/res" responses. The Client also knows how to |
| create the multicast request to that resource, by means of the Introspection Device Data hosted on |
| the Device. Therefore, the Client is able to send an UPDATE request to the multicast enabled |
| Resources. |
| The Client of an SSM Group cannot form SSM Requests for the SSM Group until the Client is |
| provisioned with the SSM Client Context for the SSM Group. Likewise, each Server in an SSM |
| Group cannot process SSM Requests for the SSM Group until the Server is provisioned with the |
| SSM Server Context for the SSM Group. The SSM Client Context and SSM Server Context are |
| provisioned by an OBT as specified in [OCF Onboarding Tool Specification.](#_bookmark8) Clause [16.3.4](#_bookmark341) specifies |
| how the OSCORE Sender Context at a Client is derived from an SSM Client Context, and how the |
| OSCORE Recipient Context at a Server is derived from an SSM Server Context. |
| The UPDATE request encapsulated in an SSM Request includes a local URI path for a target |
| Resource. A Server in the SSM Group for whom the request is intended, will process the request |
| using the Resource at this local URI path, if such a Resource exists and the Resource matches the |
| Resource Type and OCF Interface in the request. The SSM feature is designed with the |
| assumption that the local URI path, Resource Type and supported OCF Interfaces on the intended |
| Servers are consistent; but the SSM feature does not specify how such consistency is achieved. |

|  |
| --- |
| The UPDATE request message itself is expected to contain information in such way that the Server |
| can determine if the received UPDATE request message is intended for the Server, but the |
| specification of this information is not part of the SSM feature. |
| **16.3.3 OSCORE protection and verification of Simple Secure Multicast Requests** |
| All OSCORE message processing requirements in clauses 8.1 and 8.2 in [IETF RFC 8613](#_bookmark27) apply. |
| If a Client has an established SSM Client Context associated with an SSM Group, then the following |
| call flow applies whenever the Client sends a multicast non-confirmable UPDATE request targeting |
| multicast enabled Resources hosted on one or more Servers of the SSM Group. |
| 1) The Client shall apply the OSCORE request protection processing to the UPDATE request as |
| specified in clause 8.1 in [IETF RFC 8613,](#_bookmark27) using the OSCORE Security Context derived from |
| the SSM Client Context as specified in clause [16.3.4.](#_bookmark341) See [ISO/IEC 30118-1](#_bookmark3) for details on |
| setting the Proxy-URI option. |
| The Client shall send the resulting OSCORE request message to the predefined All OCF Nodes |
| multicast address. Dependent on the deployment scenario the different scopes as defined in |
| clause 12.2.9 of [ISO/IEC 30118-1](#_bookmark3) can be used. |
| 2) All Servers subscribed to the predefined multicast address receive a copy of the OSCORE |
| request message. Each Server supporting SSM which receives the OSCORE request message |
| shall apply the OSCORE request verification and decryption processing in clause 8.2 of [IETF](#_bookmark27) |
| [RFC 8613](#_bookmark27) with the following clarifications: |
| a) At Step 2 in clause 8.2 of [IETF RFC 8613](#_bookmark27) |
| i) If either the decompression or the COSE message fails to decode, the Server shall |
| ignore the message and shall not respond. |
| ii) The Server attempts to retrieve the SSM Server Contexts with "recipientID" matching |
| the ’kid’ parameter. If the Server fails to retrieve an SSM Server Context with |
| "recipientID" matching the 'kid' parameter received, then the Server shall ignore the |
| message and shall not respond. |
| b) At step 6 in clause 8.2 of [IETF RFC 8613,](#_bookmark27) if the decryption failed then the Server shall |
| ignore the message and shall not respond. |
| 3) If any of the following criteria are met, then the CRUDN request message shall be silently |
| discarded, and a response shall not be sent: |
| – The operation of the CRUDN request is not the non-confirmable UDPATE operation on a |
| multicast address. |
| – The UPDATE request message is not intended for the Server – see clause [16.3.2](#_bookmark339) for further |
| details. |
| – There is no Resource hosted on the Server at the local URI path in the UPDATE request |
| message. |
| 4) The Server shall process the UPDATE request message (encapsulated in the OSCORE request |
| message). The Server shall treat the value of "subjectuuid" in the credential entry which |
| contains the OSCORE Security Context used to verify and decrypt the OSCORE request |
| message in Step 2 as Client's Device UUID for access control processing. The Server shall |
| treat the connection type as "auth-crypt" for access control processing. The Server shall not |
| send a response. |
| The mechanism outlined is for sending a message in a send and forget mode, i.e. sending a |
| message to a group of Servers, where each Server does not acknowledge the receipt. Since |
| multicast requests are typically unreliable (e.g. non-confirmable messages) the best practice is to |
| send the same UPDATE request more than once in a short time frame. This is sufficient since the |
| multicast communication has in most cases a unicast variant for the same UPDATE request. |

|  |
| --- |
| Notification (see clause [11.3](#_bookmark185) of [ISO/IEC 30118-1](#_bookmark3)) may be used to verify if the actual UPDATE |
| request has been executed. If a subset of the group of Servers did not receive the UPDATE request, |
| unicast (confirmable) messages can be used to complete the desired overall state of the system. |
| **16.3.4 Creating OSCORE Security Context for Simple Secure Multicast** |
| The present clause specifies how |
| – a Client of an SSM Group creates a OSCORE Security Context from a SSM Client Context |
| provisioned to a credential entry of the Client. |
| – a Server of an SSM Group creates a OSCORE Security Context from a SSM Server Context |
| provisioned to a credential entry of the Server. |
| All configurable parameters of the OSCORE Security Context are either: |
| – fixed to the OSCORE-specified default value, or |
| – directly provisioned by an OBT to a credential entry of the "/oic/sec/cred" Resource. |
| The following parameters of the OSCORE Security Context used for encryption by the Client of an |
| SSM Group shall be set to the default values defined in clause 3.2 of IETF RFC 8613 (this |
| information is not configured by the OBT): |
| – AEAD Algorithm, |
| – HKDF, |
| – Master Salt, |
| – ID Context. |
| The following parameters of the OSCORE Security Context parameters used for encryption by the |
| Client of an SSM Group are derived from the SSM Client Context provisioned to a credential entry |
| of "/oic/sec/cred" of the Client: |
| – The "subjectuuid" may be any schema compliant value. This Property serves no purpose when |
| used in an SSM Client Context. |
| – The credential entry is identified as an SSM Client Context when the "credtype" matches the |
| value specified for a SSM Client Context in [Table 21,](#_bookmark224) clause [13.3.1.](#_bookmark221) |
| – The "privatedata" Property contains a 256-bit value which shall be used as the OSCORE Master |
| Secret. |
| – The OSCORE Configuration parameters ("oscore") Property is present, and includes the |
| following Properties: |
| – The "senderid" Property shall be used as the OSCORE Sender ID of the OSCORE Security |
| Context. The "recipientid" Property value shall be interpreted as the hexadecimal |
| representation of a 56-bit value. The first byte of this value is expected to have the value |
| assigned in Table Y for Simple Secure Multicast. |
| – The "desc" Property is not used in security processing. This Property is described in clause |
| [9.3.9.](#_bookmark148) |
| On the Device shutting down, for each such credential entry, the Device shall write the value of |
| corresponding OSCORE Sender Sequence Number as "ssn" Property to non-volatile memory. In |
| event of a crash, devices should apply Appendix B.1.1 of [IETF RFC 8613.](#_bookmark27) |
| The following parameters of the OSCORE Security Context used by a Server of an SSM Group for |
| verification and decryption shall be set to the default values defined in clause 3.2 of [IETF RFC](#_bookmark27) |
| [8613](#_bookmark27) (this information is not configured by the OBT): |
| – AEAD Algorithm, |
| – HKDF, |

|  |
| --- |
| – Replay Window, |
| – Master Salt, |
| – ID Context. |
| The following parameters of the OSCORE Security Context parameters used by a Server of an |
| SSM Group for verification and decryption are derived from the SSM Server Context provisioned |
| to a credential entry of "/oic/sec/cred" of the Server: |
| – The "subjectuuid" is used for access control processing as described in Step 4 of clause [16.3.3.](#_bookmark340) |
| – The credential entry is identified as an SSM Server Context when the "credtype" matches to |
| the value specified for an SSM Server Context in [Table 21,](#_bookmark224) clause [13.3.1.](#_bookmark221) |
| – The "privatedata" Property of the credential entry contains a 256-bit value which shall be used |
| as the OSCORE Master Secret. |
| – The OSCORE Configuration parameters ("oscore") Property is present, and includes the |
| following Properties: |
| – The "recipientid" Property shall be used as the OSCORE Recipient ID of the OSCORE Security |
| Context. The "recipientid" Property value shall be interpreted as the hexadecimal representation |
| of a 56-bit value. The first byte of this value is expected to have the value assigned in Table Y |
| for Simple Secure Multicast. |