ACE Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 21, 2016

S. Gerdes
Universitaet Bremen TZI

October 19, 2015

Server-Initiated Ticket Request
draft-gerdes-ace-dcaf-sitr-00

Abstract

The Delegated CoAP Authorization Framework (DCAF) defines how constrained devices can securely obtain security associations and authorization information from their respective less constrained devices, the Authorization Managers. In DCAF a constrained client requests an authorization ticket from the Server Authorization Manager (SAM) by contacting its own Client Authorization Manager (CAM). However, there may be cases where this approach is not applicable, e.g., because the client is not able to reach Authorization Managers in the Internet.

Specifically for these situations, this document defines the Server-Initiated Ticket Request (SITR) that specifies how a constrained server can request authorization tokens and securely obtain security associations and authorization information for mutual authenticated authorization with the client.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 21, 2016.

Table of Contents

1. Introduction 2
1.1. Terminology 2
2. Protocol 3
2.1. Overview 3
2.2. CAM Information Message 4
2.3. Server-Initiated Access Request Message 5
2.4. Server-Initiated Ticket Request Message 5
2.5. SI Ticket Grant Message 6
2.6. SI Ticket Transfer Message 7
2.7. CAM Information Response 7
3. Payload Format 8
4. Content Types 8
5. IANA Considerations 8
6. Security Considerations 8
7. Acknowledgments 9
8. References 9
8.1. Normative References 9
8.2. Informative References 9
Author's Address 9

1. Introduction

See abstract.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Readers are required to be familiar with the terms and concepts defined in [I-D.ietf-ace-actors] and [I-D.gerdes-ace-dcaf-authorize].

2. Protocol

2.1. Overview

The figure Figure 1 depicts the Sitr protocol flow:

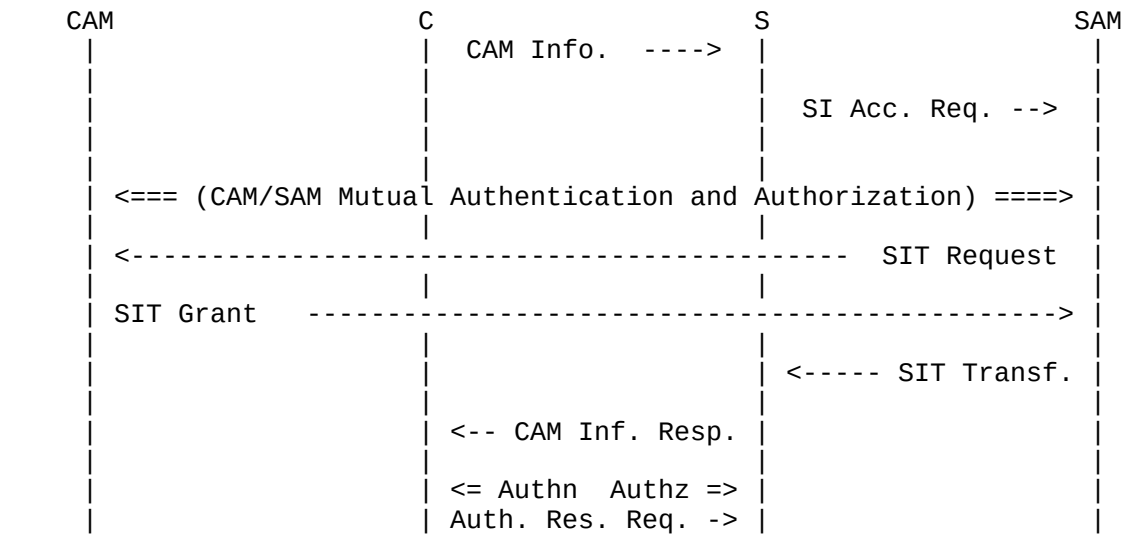

Figure 1: Protocol Overview

The authorization flow will then look as follows: C will send a CAM info message (maybe after first sending an unauthorized request to S that is denied) that contains its CAM address together with the request to this URI.

S will send a Server Initiated (SI) Access Request to SAM which includes the request and URI. SAM will contact CAM and determine if CAM has the respective permissions. The details of the communication between CAM and SAM are not in scope for this document, but CAM and SAM will mutually authenticate each other and then initiate a secure communication. Then SAM sends a SIT Request to CAM which contains the information from the Resource Request.

CAM checks if SAM is authorized according to COP’s authorization policies (mutual authorization). If this is the case, CAM creates a SI ticket. The ticket contains keying material for the secure communication between C and S and, if necessary, authorization

information that reflect COP's security policies for C. CAM sends this ticket to SAM, SAM includes server authorization information to the SI ticket if necessary and sends the ticket to S. S keeps one part of the ticket and sends the other part to C as a reply to C's CAM Information message. With their respective part of the ticket, C and S can communicate securely.

2.2. CAM Information Message

C sends a CAM Information message to S to stimulate S to request a ticket for C. The message is constructed as follows:

1. The request method is FETCH (see [I-D.bormann-core-coap-fetch]).

2. The request URI is set to the URI of the requested resource.

3. The message payload contains a data structure that describes the action and resource for which C requests an access ticket as well as the CAM URI.

The data structure in the payload MUST contain:

1. The contact information for the CAM to use: a URI that specifies the CAM in charge of C.

2. A URI of the resource that C wants to access.

3. The actions that C wants to perform on the resource.

Figure 2 shows an example for a CAM Information message. (Refer to Section 3 for a detailed description of the available attributes and their semantics.)

```
FETCH /s/tempC
Content-Format: application/dcaf+cbor
{
  CAM: "coaps://sam.example.com/authorize",
  SAI: ["coaps://temp451.example.com/s/tempC", 5],
  TS: 168537
}
```

Figure 2: CAM Info Message Example

Note: if used with object security, the FETCH request contains CBOR encoded message syntax structure (COSE), that conveys the application/dcaf+cbor payload.

The example shows a CAM information message payload for the resource "/s/tempC" on the Server "temp451.example.com". Requested operations in attribute SAI are GET and PUT.

The response to a CAM Information message is delivered by S back to C in a CAM Information Response message.

2.3. Server-Initiated Access Request Message

A server that receives a CAM Information message MAY use the information in the payload of the message to request a Server-Initiated (SI) Ticket for C. To do so, it contacts its own SAM. The SI Access Request is constructed as follows:

1. The request method is POST.

2. The request URI is set as described below.

3. The message payload contains a data structure that describes the action and resource for which C requests an access ticket as well as the CAM URI.

The request URI identifies a resource at SAM for handling authorization requests from C. The URI SHOULD be announced by SAM in its resource directory as described in section 9 of [I-D.gerdes-ace-dcaf-authorize].

The message payload is constructed from the information that C has sent in its CAM Information message (see Section 2.2). The request MUST contain the attributes described in Section 2.2.

2.4. Server-Initiated Ticket Request Message

When SAM receives a Server-Initiated Access Request message from S and ROP specified authorization policies for S, SAM MUST check if the requested actions are allowed according to these policies. If all requested actions are forbidden, SAM MUST send a 4.03 response.

If no authorization policies were specified or some or all of the requested actions are allowed according to the authorization policies, SAM either returns a cached response or attempts to create a SI Ticket Request message. The SI Ticket Request message MAY contain all actions requested by C since SAM will add SAI in the Ticket Transfer Message if ROP specified authorization policies (see Section 2.6).

SAM MAY return a cached response if it is known to be fresh according to Max-Age. SAM SHOULD NOT return a cached response if it expires in less than a minute.

If CAM does not send a cached response, it checks the content type of the request payload and validates that the payload contains at least the fields CAM and SAI. SAM MUST respond with 4.00 (Bad Request) if the type does not belong to the allowed content-types and if any of these fields is missing or does not conform to the format described in Section 3.

If the payload is correct, SAM creates a SIT Request message from the SI Access Request received from S as follows:

1. The destination of the Ticket Request message is derived from the "CAM" field that is specified in the Access Request message payload (for example, if the SI Access Request contained 'CAM: "coaps://cam.example.com/authz"', the destination of the Ticket Request message is cam.example.com).

2. The request method is POST.

3. The request URI is constructed from the CAM field received in the Access Request message payload.

4. The payload is copied from the SI Access Request sent by S.

CAM and SAM MUST be able to mutually authenticate each other, e.g. based on a public key infrastructure and MUST be able to communicate securely.

2.5. SI Ticket Grant Message

When CAM has received a SI Ticket Request message it has to evaluate the access request information contained therein. First, it checks whether the request payload is of a supported content type (see Section 4) and contains at least the fields CAM and SAI. CAM MUST respond with 4.00 (Bad Request) for CoAP (or 400 for HTTP) if any of these fields are missing or do not conform to the format described in Section 3.

CAM decides whether or not access is granted to the requested resource and then creates a SI Ticket Grant message that reflects the result. CAM initializes the access ticket comprised of a Face and the Client Information (CI).

The CI contains:

- o the Client authorization information (CAI)
- o a nonce

CI MAY additionally contain:

- o a lifetime
- o a CAI identifier (for revocation)
- o keying material for C (if no key derivation method is used to generate the verifier in Face)

The Face at this point only comprises the verifier. CAM MAY generate the verifier using the method described in section 6.2 of [I-D.gerdes-ace-dcaf-authorize]. CAM MUST NOT include Server Authorization information (SAI) in the ticket Face.

The CI MUST be integrity-protected on the way to C. CAM MAY additionally protect the confidentiality of CI. If the CI contains keying material, CAM MUST ensure the confidentiality of CI.

The confidentiality of Face MUST be ensured on the way to SAM.

The SI Ticket Grant messages is then constructed as a success response (2.05 for CoAP, 200 for HTTP) with the ticket as content.

2.6. SI Ticket Transfer Message

The SI Ticket Transfer message is the response to the SI access request and delivers the ticket to S.

The CAI provided by CAM in the SI Ticket Grant message provide only the client-side permissions. If ROP defined access permissions for S, SAM MUST add server authorization information (SAI) to Face that reflect those policies. SAM MUST NOT include SAI that were provided by CAM.

SAM MUST provide for the confidentiality and integrity of Face when transmitting it to S. SAM MAY encrypt the CI.

2.7. CAM Information Response

When S receives a SI Ticket Transfer message, it MUST make sure that it contains the Face and the CI. If Face contains SAI, S MUST validate its authenticity and integrity. S keeps the ticket Face and sends the CI to C. S MAY transmit the answer to C’s initial request provided in the CAM Info message together with the CI.

When C receives the CAM Information Response, it MUST validate that the CI was generated by CAM and not modified. With the information in the CI, C can start a secure communication with S.

C MAY establish a security context with S using the verifier provided in the CI, e.g., by initiating a DTLS session with the verifier as the Pre-shared Key.

3. Payload Format

SITR uses the CBOR representation defined in DCAF (see section 5 of [I-D.gerdes-ace-dcaf-authorize]) and additionally defines a CBOR value for CAM:

Encoded Value	Key
13	CAM

Table 1: SITR field identifiers encoded in CBOR

4. Content Types

The supported content types are:

- "application/dcaf+cbor"
- "application/cose+cbor"

5. IANA Considerations

None

6. Security Considerations

For solutions where the server requests the ticket for the client, most of the workload (send a message to the authorization manager, wait for the answer, keep state in the meantime) is on the server which makes it susceptible to DOS attacks. Therefore, as with all solutions state based on client requests, these solutions MUST NOT be used except in conjunction with appropriate mitigation. Where applicable, it is recommended to use DCAF instead, where the client has to request the ticket.

7. Acknowledgments

The author would like to thank Bert Greevenbosch, Olaf Bergmann and Carsten Bormann for their valuable input and feedback.

8. References

8.1. Normative References

[I-D.bormann-core-coap-fetch]
Bormann, C., "CoAP FETCH Method", draft-bormann-core-coap-fetch-00 (work in progress), October 2015.

[I-D.gerdes-ace-dcaf-authorize]
Gerdes, S., Bergmann, O., and C. Bormann, "Delegated CoAP Authentication and Authorization Framework (DCAF)", draft-gerdes-ace-dcaf-authorize-03 (work in progress), September 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

8.2. Informative References

[I-D.ietf-ace-actors]
Gerdes, S., Seitz, L., Selander, G., and C. Bormann, "An architecture for authorization in constrained environments", draft-ietf-ace-actors-02 (work in progress), October 2015.

Author’s Address

Stefanie Gerdes
Universitaet Bremen TZI
Postfach 330440
Bremen D-28359
Germany

Phone: +49-421-218-63906
Email: gerdes@tzi.org