ACE Working Group
Internet-Draft
Intended status: Informational
Expires: December 1, 2014

S. Gerdes
Universitaet Bremen TZI

May 30, 2014

# Actors in the ACE Architecture
## draft-gerdes-ace-actors-00

Abstract

Constrained nodes are small devices which are limited in terms of processing power, memory, non-volatile storage and transmission capacity. Due to these constraints, commonly used security protocols are not easily applicable. Nevertheless, an authentication and authorization solution is needed to ensure the security of these devices.

This document defines actors in the security architecture for authentication and authorization, analyzes the relationships between them, and describes their respective tasks and characteristics. This knowledge will then be used to derive requirements for the communication between the actors.

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 December 1, 2014.

Table of Contents



## 1. Introduction

Constrained nodes are small devices with limited abilities which in many cases are made to fulfill a single simple task. They have limited system resources such as processing power, memory, non-volatile storage and transmission capacity and additionally in most cases do not have user interfaces and displays. Due to these constraints, commonly used security protocols are not always easily applicable.

Constrained nodes are expected to be integrated in all aspects of every day live and thus will be trusted with a lot of personal data. Without appropriate security mechanisms attackers might gain control over things relevant to our lives. Authentication and authorization mechanisms are therefore prerequisites for a secure Internet of Things.

The Authentication and Authorization in Constrained Environments (ACE) Working Group aims at defining a solution for authenticated and authorized access to resources.

To achieve this, it is necessary to develop a deep understanding about the problem to be solved. An essential part of this is to identify the various "players" in the scenario: What are the relevent actors in the archicture and which tasks do they fulfill? How can the relationships between the actors be defined?

This document defines actors, their relationships and resulting security requirements for authentication and authorization in constrained environments.

## 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].

This document uses the following terminology:

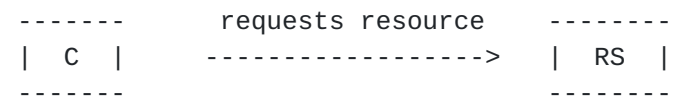
Resource: an item of interest. It might contain sensor or actuator values or other information. The author had resources in the sense of RFC2616 [RFC2616] in mind, but for the considerations in this document the kind of representation of the item is not relevant.

Constrained node: a constrained device in the sense of [RFC7228].

# 2. Problem Statement

The problem the ACE Working Group aims to solve can be summarized as follows:

- A Client (C) wants to access a Resource (R) on a Resource Server (RS).
- A priori, C and RS do not know each other and have no trust relationship.
- C and / or RS are constrained.



There are some security requirements for this scenario including one or more of:

- No unauthorized entity must be able to access (or otherwise gain knowledge of) R.

- C must access the proper R.

Therefore, RS needs to know if C is allowed to access R and if that is the case needs to make sure that it provides the resource only to C. C needs to know if R as offered by RS really is the resource it wants to access.

## 3. Actors

This section describes the various actors in the architecture. An actor is identified by the tasks it has to fulfill. Several actors might share a single device or even be combined in a single piece of software. Interfaces between actors may be realized as protocols or be internal to such a piece of software.

### 3.1. Constrained Level Actors

As described above, either C or RS or both of them may be located on a constrained node. Although they are not necessarily constrained they should be able to operate if they are. We therefore derive from the problem description that C and RS must be able to perform their tasks even if they are located on a constrained node. Thus, C and RS are considered to be Constrained Level Actors.

RS is hosting a resource R.

R is an item of interest such as a sensor or actuator value. R is considered to be part of RS and is not a separate actor. The device on which RS is located might contain several resources of different resource owners. For simplicity of exposition, these resources are described as if they had separate RS.

C attempts to access a resource on RS.

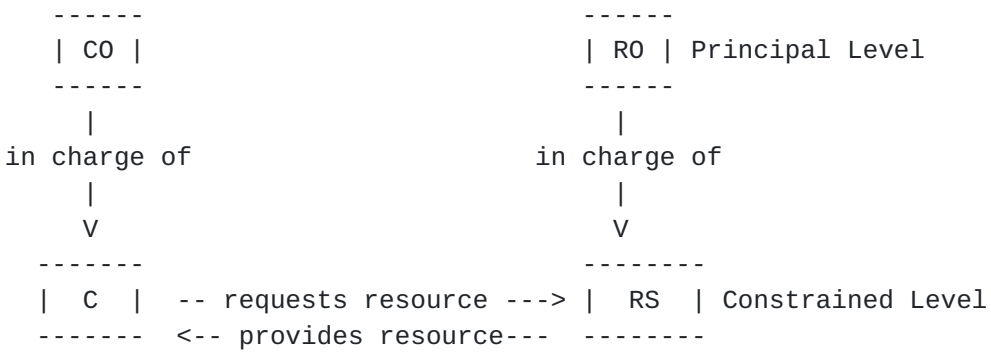
As C and RS do not previously know each other they might belong to different security domains.

### 3.2. Principal Level Actors

Our objective is that C and RS are under control of principals in the physical world, the Client Owner (CO) and the Resource Owner (RO) respectively. The owners decide about the security policies of their respective devices and belong to the same security domain.

CO is in charge of C, i.e. CO specifies security policies for C, e.g. with whom C is allowed to communicate. By definition, C and CO belong to the same security domain.

RO is in charge of R and RS. RO specifies authorization policies for R and decides with whom RS is allowed to communicate. By definition, R, RS and RO belong to the same security domain.

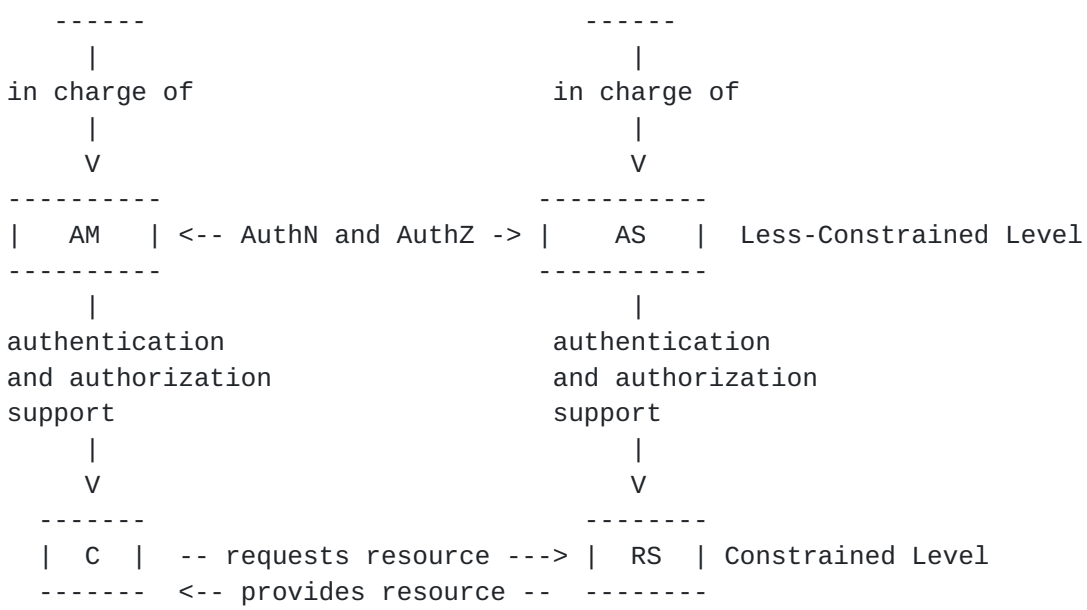



### 3.3. Less-Constrained Level Actors

Constrained level actors can only fulfill a limited number of tasks and may not have network connectivity all the time. To relieve them from having to manage keys for numerous devices and conducting computationally intensive tasks, another complexity level for actors is introduced. An actor on the less-constrained level belongs to the same security domain as its respective constrained level actor. They also have the same principal.

The Authentication Manager (AM) belongs to the same security domain as C and CO. AM acts on behalf of CO. It is aiding C in the authentication of RS and determining if RS is an authorized source for R.

The Authorization Server (AS) belongs to the same security domain as R, RS and RO. AS acts in behalf of RO. It supports RS by authenticating C and determining C's permissions on R.




For a more detailed graphic please consult the PDF version.

## 4. Protocol Requirements

Devices on the less-constrained level are more powerful than constrained level devices. This results in different requirements for the protocols used on these levels.

### 4.1. Constrained Level Protocols

A protocol is considered to be on the constrained level if it is used between the actors C and RS which are considered to be constrained (see Section 3.1). C and RS might not belong to the same security domain. Therefore, constrained level protocols are required to work between different security domains.

Commonly used Internet protocols can not in every case be applied to constrained environments. In some cases, tweaking and profiling is required. In other cases it is beneficial to define new protocols which were designed with the special characteristics of constrained environments in mind.

On the constrained level, protocols must be used which address the specific requirements of constrained environments. The Constrained Application Protocol (CoAP) [I-D.ietf-core-coap] should be used as transfer protocol if possible. CoAP defines a security binding to Datagram Transport Layer Security Protocol (DTLS) [RFC6347]. Thus, DTLS should be used for channel security.

Constrained devices have only limited storage space and thus cannot store large numbers of keys. This is especially important because constrained networks are expected to consist of thousands of nodes. Protocols on the constrained level should keep this limitation in mind.

#### 4.1.1. Cross Level Support Protocols

Protocols which operate between a constrained device on one side and the corresponding less constrained device on the other are considered to be (cross level) support protocols. Protocols used between C and AM or RS and AS are therefore support protocols.

Support protocols must consider the limitations of their constrained endpoint and therefore belong to the constrained level protocols.

### 4.2. Less-Constrained Level Protocols

A protocol is considered to be on the less-constrained level if it is used between the actors AM and AS. AM and AS might belong to different security domains.

On the less-constrained level, HTTP [RFC2616] and Transport Layer Security (TLS) [RFC5246] can be used instead of CoAP and DTLS. Moreover, existing security solutions for authentication and authorization such as the Web Authorization Protocol (OAuth) [RFC6749] and Kerberos [RFC4120] can likely be used without modifications and there are no limitations for the use of a Public Key Infrastructure (PKI).

## 5. IANA Considerations

None

## 6. Security Considerations

This document discusses security requirements for the ACE architecture.

## 7. Acknowledgments

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

## 8. References

### 8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, May 2014.

## 8.2. Informative References

[I-D.ietf-core-coap] Shelby, Z., Hartke, K., and C. Bormann, "Constrained Application Protocol (CoAP)", draft-ietf-core-coap-18 (work in progress), June 2013.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC4120] Neuman, C., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

[RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.

## Author's Address

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

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