Authorization for Constrained RESTful Environments (ACRE)
draft-seitz-ace-core-authz-00

Ludwig Seitz (ludwig@sics.se)
Göran Selander (goran.selander@ericsson.com)
Mališa Vučinić (malisa.vucinic@st.com)

IETF ACE WG meeting
July 22, 2015
Example A: Access to an Actuator

- RS needs to know C is authorized
- C needs to know that the response is from RS
- Integrity and replay protection for Request/Response
- Possibly encryption for Request/Response
Example B: Multiple devices accessing Sensor Data

- Access to sensor readings must be controlled
- Clients need to be able to verify the origin of a sensor reading and to detect replay
- Other example: Access to wireless sensor network data behind application layer gateway
This draft

• Covers additional use cases
  – Supports multiple authorization schemes
  – End-to-end security with intermediaries
  – DTLS or COSE for protecting information flows
  – Using REST

• Is inspired by existing work
  – draft-selander-core-access-control
  – draft-gerdes-ace-dcaf-authorize
  – draft-bormann-core-ace-aif
  – OSCAR (object security architecture for IoT)
ACE Architecture and Information Flows

Legend:
- Black boxes represent functions
  - Functions may be combined in one node
- Information flows in solid lines
  - Resource access (based on CoAP)
  - Control information (authorization information, keys, etc.)
  - Information flow may pass intermediary nodes

Information flows may be protected with session-based security (DTLS) or data object based security (COSE)

Source: draft-gerdes-ace-actors
Authorization Schemes

**Push**

1. AuthZ request
2. AuthZ token
3. Token & Request
4. Response

**Pull**

1. Request
2. AuthZ request
3. AuthZ decision
4. Response

**Client-Pull**

1. GET Request
2. Protected resource Representation
3. AuthZ request
4. Access keys
RESTful Authorization Resources

• C or RS → AS: Authorization Request
  – POST request to authorization resource at AS
• C → RS: Authorization Info
  – POST authorization information to
    authorization resource at RS
• Different authorization schemes re-use
  the same RESTful building blocks
• Need to define discovery mechanisms
  – How about /.well-known/authorize ?
  – rt for Resource Directory ?
ACRE Design Principles

1. Allow security at different layers
   - Session-based security or object security for each information flow depending on use case

2. Allow different authorization schemes
   - Requires support for different order of information flow

3. Use REST
   - Authorization information as RESTful resources
Thank you!

Questions/comments?