Internet of Things Workshop
ST – 2015/2016

CoAP introduction

Johan Lukkien

John Carpenter, 1982
Key IoT Standardization

• IETF
  ✓ 6LoWPAN Working Group (IPv6 anywhere)
  ✓ ROLL (Routing Over Low-power Lossy Networks) WG
  ✓ CoRE WG (REST for IoT, CoAP, Resource Directory etc.)
  ✓ TLS WG (DTLS)

• OMA
  ✓ Lightweight M2M Enabler Standard (CoAP/DTLS based)
  ✓ Device Management 2.0 Enabler Standard (HTTP/TLS based)

• ETSI / OneM2M
  ✓ Ongoing work on M2M system standardization (CoAP, HTTP binding)

• W3C
  ✓ Efficient XML Interchange (EXI) standardization

• ZigBee IP
  ✓ An open-standard 6LoWPAN stack for e.g. Smart Energy 2.0
CoAP – constrained application protocol

• CoRE WG in EITF

• Documents
  – RFC 7252
  – draft-ietf-core-observe-16 (OBSERVE option)

• Authors:
  – Z. Shelby (Sensinode, ARM)
  – K. Hartke, C. Bormann (Uni Bremen TZI)

• Presentation based on web sources by Shelby
Layering provides two interfaces

- An API to develop applications, typically as a library
- The protocol between library instances
  - to be implemented by the library
- Libraries may be different
- Example:
  - CoAP protocol
  - Californium library
CoAP basics

• An efficient RESTful protocol
• UDP binding – CoAP is UDP payload
  – port 5683
• We discuss the protocol: message format, interaction and quality
**Definition: protocol**

- **Protocol:** A formal set of rules that dictates how information exchange as well as interaction between objects (can be devices, execution threads, etc.) should take place.

  The rules specify
  - the format of the messages exchanged;
  - a number of different protocol states and what messages are allowed to be sent in each state; these states determine, among others, the order of the messages.
  - timing constraints and other quality properties, if any.

- **Note:**
  - ‘message’ includes function calling
  - one may specify a protocol without being explicit about the overall service it realizes

- **Examples:** HTTP, TCP, UDP, SOAP, DNS
Overview

- Requirements
- Transactions
- Methods
- Response
- Message identification
- Options
- Resource Discovery
CoAP – requirements

• Limited resources
  – small footprint
  – constrained networks
  – sleeping nodes – proxying

• Reliability under application control
  – transaction semantics moved from transport protocol to CoAP

• Support for Internet REST operation
  – caching
  – mapping to HTTP
  – resource manipulation

• Machine to machine operation
  – resource discovery
  – publish / subscribe / notify
  – multicast
The CoAP Architecture

From presentation Shelby: CoAP: the IoT protocol
Naming on the Internet

- Name: identity
- Locator: address

Universal Resource Identifier (URI)

Universal Resource Name (URN)

```
urn:Sensei:sensinode.com:NanoSensor:N740:3a-43-ff-12-01-01
```

Universal Resource Locator (URL)

```
http://www.example.org:8080/sensors?id=light1
```

From presentation Shelby: CoAP: the IoT protocol

coop://example.tue.nl:5683/sensors/light1
coaps://example.tue.nl:5683/sensors/light1
Transactions

**Types**

- **CON**
  - confirmable, respond with ACK
- **NON**
  - non-confirmable
- **ACK**
  - acknowledge CON
- **RST**
  - deny (resete) interaction

**Parameters**

- **Message ID** (transaction ID)
- **Code**
  - Request (method) or Response
- **Optional numbered options**, e.g.,
  - URI
  - media
- **Optional token**
  - ‘interaction ID’
- **(Payload)**
Acknowledgement versus Response

- An ACK confirms receipt of a CON
- A response gives the result of a request
  - transmitted using CON / ACK again
  - may piggy-back on the first ACK

Client  
CON [0xbc90]  
GET /temperature  
(Token 0x71)  

ACK [0xbc90]  
2.05 Content  
(Token 0x71)  
"22.5 C"

Server  

--- Time Passes ---

Client  
CON [0x23bb]  
2.05 Content  
(Token 0x73)  
"22.5 C"

ACK [0x23bb]  

Server  
CON [0x7a10]  
GET /temperature  
(Token 0x73)  

ACK [0x7a10]  

---
Requests and Responses

Request methods

- **GET URI**
  - retrieve (resource identified by) URI
- **PUT URI**
  - update URI
- **DELETE URI**
  - delete URI
- **POST URI**
  - perform operation and create new resource under URI
- GET is safe, GET, PUT and DELETE are idempotent

Responses

- **Class 2 (success)**
  - 2.01: created, 2.02: deleted, 2.03: valid, 2.04: changed, 2.05: content
- **Class 4 (client error)**
  - 4.00: bad request, 4.01: unauthorized, 4.02: bad option, 4.03: forbidden, 4.04: not found, 4.05: method not allowed, 4.06: not acceptable, 4.12: precondition failed
- **Class 5 (server error)**
  - 5.00: internal server error, 5.01: not implemented, 5.02: bad gateway, 5.03: service unavailable, 5.04: gateway timeout, 5.05: proxying not supported
The packet format

- Version
- Type
- Token Length
- (Request or Reply) Code
- 0xFF: payload marker
From presentation Shelby: CoAP: the IoT protocol
<table>
<thead>
<tr>
<th>No.</th>
<th>C</th>
<th>U</th>
<th>N</th>
<th>R</th>
<th>Name</th>
<th>Format</th>
<th>Length</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>If-Match</td>
<td>opaque</td>
<td>0-8</td>
<td>(none)</td>
</tr>
<tr>
<td>3</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Uri-Host</td>
<td>string</td>
<td>1-255</td>
<td>(see below)</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td>ETag</td>
<td>opaque</td>
<td>1-8</td>
<td>(none)</td>
</tr>
<tr>
<td>5</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>If-None-Match</td>
<td>empty</td>
<td>0</td>
<td>(none)</td>
</tr>
<tr>
<td>7</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Uri-Port</td>
<td>uint</td>
<td>0-2</td>
<td>(see below)</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>Location-Path</td>
<td>string</td>
<td>0-255</td>
<td>(none)</td>
</tr>
<tr>
<td>11</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Uri-Path</td>
<td>string</td>
<td>0-255</td>
<td>(none)</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Content-Format</td>
<td>uint</td>
<td>0-2</td>
<td>(none)</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>Max-Age</td>
<td>uint</td>
<td>0-4</td>
<td>60</td>
</tr>
<tr>
<td>15</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Uri-Query</td>
<td>string</td>
<td>0-255</td>
<td>(none)</td>
</tr>
<tr>
<td>17</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>Accept</td>
<td>uint</td>
<td>0-2</td>
<td>(none)</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>Location-Query</td>
<td>string</td>
<td>0-255</td>
<td>(none)</td>
</tr>
<tr>
<td>35</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Proxy-Uri</td>
<td>string</td>
<td>1-1034</td>
<td>(none)</td>
</tr>
<tr>
<td>39</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>Proxy-Scheme</td>
<td>string</td>
<td>1-255</td>
<td>(none)</td>
</tr>
<tr>
<td>60</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>Size1</td>
<td>uint</td>
<td>0-4</td>
<td>(none)</td>
</tr>
</tbody>
</table>

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable

Classes encoded in option number
The OBSERVE option

• In GET:
  – obtain current resource representation
  – add/remove to list of observers (value 0/1)

• As a response:
  – interpret as notification
  – value is sequence number
<table>
<thead>
<tr>
<th>t</th>
<th>Observed State</th>
<th>CLIENT</th>
<th>SERVER</th>
<th>Actual State</th>
<th>Actual State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>unknown</td>
<td>+------&gt;</td>
<td>18.5 Cel</td>
<td>Header: GET 0x41011633</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Get</td>
<td></td>
<td></td>
<td>Token: 0x4a</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>Uri-Path: temperature</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>Observe: 0 (register)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>&lt;-------</td>
<td></td>
<td>Header: 2.05 0x61451633</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td>2.05</td>
<td>Token: 0x4a</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>18.5 Cel</td>
<td></td>
<td></td>
<td>Observe: 9</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td>Max-Age: 15</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td>Payload: &quot;18.5 Cel&quot;</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>&lt;-------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td>2.05</td>
<td>Header: 2.05 0x51457b50</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>19.2 Cel</td>
<td></td>
<td></td>
<td>Token: 0x4a</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td>Observe: 16</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td>Max-Age: 15</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td>Payload: &quot;19.2 Cel&quot;</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Options within message field

- **Delta**: offset from previous number
  - hence, options listed in increasing order
  - 13,14,15 reserved for extending

- **Length**: of value
  - 13,14,15 reserved for extending
Caching

• Motivation
  – reduce network load
  – act on behalf of low resource (sleeping) node

• Cacheability derives from response code
  – key determined by request and options that are selected as cache key

• Max-Age option determines validity

• Validation through Etag
Example

From presentation Shelby: CoAP: the IoT protocol
# Configuration parameters

<table>
<thead>
<tr>
<th>name</th>
<th>default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK_TIMEOUT</td>
<td>2 seconds</td>
</tr>
<tr>
<td>ACK_RANDOM_FACTOR</td>
<td>1.5</td>
</tr>
<tr>
<td>MAX_RETRANSMIT</td>
<td>4</td>
</tr>
<tr>
<td>NSTART</td>
<td>1</td>
</tr>
<tr>
<td>DEFAULT_LEISURE</td>
<td>5 seconds</td>
</tr>
<tr>
<td>PROBING_RATE</td>
<td>1 byte/second</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>name</th>
<th>default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX_TRANSMIT_SPAN</td>
<td>45 s</td>
</tr>
<tr>
<td>MAX_TRANSMIT_WAIT</td>
<td>93 s</td>
</tr>
<tr>
<td>MAX_LATENCY</td>
<td>100 s</td>
</tr>
<tr>
<td>PROCESSING_DELAY</td>
<td>2 s</td>
</tr>
<tr>
<td>MAX_RTT</td>
<td>202 s</td>
</tr>
<tr>
<td>EXCHANGE_LIFETIME</td>
<td>247 s</td>
</tr>
<tr>
<td>NON_LIFETIME</td>
<td>145 s</td>
</tr>
</tbody>
</table>