M2M, IOT, DEVICE MANAGEMENT
COAP AND LIGHTWEIGHT M2M TO RULE THEM ALL?

Julien Vermillard - Sierra Wireless
Eclipsecon France 2014
WHO AM I?

Software Engineer at Sierra Wireless, implementing various protocols for AirVantage cloud service

Apache Software Foundation member
Eclipse committer on Californium and Wakaama
AGENDA

M2M/IoT Protocols: MQTT, CoAP, XMPP

Device management: TR-069, OMA-DM, LWM2M

Impact on IoT architectures
M2M/IoT APPLICATIONS

My definition:

Large fleet of cloud connected devices solving a business problem

(Ex: oil pump monitoring, truck fleet tracking)
VERSATILE PROTOCOLS

Usual suspects:

- XMPP
- MQTT
- COAP
MQ TELEMETRY TRANSPORT

Very simple and light protocol on top of TCP

Good fit for wireless applications

Publish/Subscribe paradigm

Websocket support

Centralized
A sensor pushes telemetry values on some topics:

- greenhouse/42/temperature
- greenhouse/42/humidity
- greenhouse/42/luminosity

Actions are on another topic:

- greenhouse/42/open-the-roof

Payload format is free (json, binary, whatever..)
MQTT @ ECLIPSE IOT

**Paho**: for clients (Java, C/C++, Python, Js, Go, Lua)

**Mosquitto**: light and full feature broker

**Moquette**: Java broker with websockets
Ponte: Node.js server bridging MQTT, HTTP, CoAP

Kura: M2M application framework with MQTT as default transport
WHY NOT HTTP?

We all know it or not...

Useless text headers

Verbose

Polling
COAP:
Constrained Application Protocol
Internet Eng. Task Force standard for Internet of things
Started in 2010!
RFC 7252
Simple to encode/decode: targets 8bit MCU

UDP based, targets low power IP networks: 6LowPAN

Two level of QoS: confirmable messages or not
COAP: RESTFUL THINGS!

REST paradigm for things:

```
coap://myhouse.local/lamps/7/status
```

HTTP like verbs: **GET, POST, PUT, DELETE**

Content negotiation

Web goodies: but in a compact binary format!
<table>
<thead>
<tr>
<th>Ver</th>
<th>T</th>
<th>TKL</th>
<th>Code</th>
<th>Message ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Token (if any, TKL bytes) ...
| Options (if any) ...
| Payload (if any) ...

---
COAP: DISCOVERABILITY

GET coap://hostname/.well-know/core

Provides a list of all supported resources

</config/groups>;rt="core.gp";ct=39,
</sensors/temperature>;rt="ucum.Cel";ct="0 41 50";obs,
</large>;rt="block";sz=1280;title="Large resource"
COAP: BONUS

Observe for streaming changing values

Local group communication (multicast)

HTTP <=> CoAP proxy

Web like Caching

Resource directory

All the Web model for IoT: Web of Things
**COAP: SECURITY**

**DTLS:** TLS on Datagram

Works with PSK (pre-share-key) or certificate chains

**Here be dragons!**

The real implementation complexity
COAP @ ECLIPSE IOT

Californium: Java Coap server and client (with DTLS)

Ponte: again :) since it's bridging everything
DEVICE MANAGEMENT

Definition:
Secure, monitor, manage large fleet of deployed devices
DEVICE MANAGEMENT

Configure the device

Update the firmware (and maybe the app)

Monitor and gather connectivity statistics
INTEROPERABILITY IS THE KEY

you don't know yet what hardware will power your IoT projects on the field,

but you MUST be able to do device management in a consistent way
D.M. PROTOCOLS

Usual suspects:

- TR-069
- OMA-DM
- Lightweight M2M
SOAP API for broadband modems

Yes SOAP!
An Open Mobile Alliance standard for Device Management

Targets mobile phone terminals but can be used for M2M

Meant to be used by mobile network operators
OMA-DM FEATURES

Read, write configuration or monitoring nodes

Trigger remote commands (Exec)

FUMO:
Firmware Update Management Object

SCOMO:
Software Component Management Object
OMA-DM

HTTP/XML based, with a binary XML (WBXML) encoding

Weird phone features gets in the way: every communication the device gives its language (ex: EN_en)

Binary SMS for wakeup and bootstraping
OMA-DM SECURITY

HMAC MD5: HTTP Header signing the payload

Use HTTPS if you need confidentiality
OMA LIGHTWEIGHT M2M

A new Open Mobile Alliance standard

An OMA-DM successor for M2M targets
LWM2M: COAP

Built on top of CoAP:

Really lighter than OMA-DM and TRS-069
LWM2M FEATURES

Firmware upgrades (in band or thru http)

Device monitoring and configuration

Server provisioning (bootstrapping)
LWM2M SMS

SMS can be used for waking-up the device.

Or any GET/POST/PUT/DELETE.

The device can reply by SMS or UDP (return-path).
LWM2M: STANDARDS OBJECTS

Device
Server
Connectivity monitoring
Connectivity statistics
Location
Firmware

The objects have a numerical identifier.
LWM2M THE URLS

/{object}/{instance}/{resource}

Examples:

"/6/0" the whole position object (binary record).

"/6/0/2" only the altitude.
LWM2M @ ECLIPSE IOT

Wakaama (ex liblwm2m)

A C library for implementing LWM2M in your devices.

Embedded friendly.
MORE LWM2M OPENSOURCE

A Java based LWM2M server and library.

Based on Californium (Eclipse).

http://github.com/jvermillard/leshan
DM/Apps: How everything fall into place?
ARCHITECTURE OF YOUR APPLICATION

Embedded Application

Cloud servers

MQTT or CoAP for fun and profits!
A MORE REALISTIC ARCHITECTURE

- O/S
- Supervisor
- Application

OMA-DM

MQTT or CoAP

Cloud servers
BUT IN FACT...

Cloud servers

Network Operator

OMA-DM

OMA-DM

OMA-DM

MQTT/CoAP

Radio module
2G/3G/LTE
ZigBee/6LoWPAN
WiFi, etc.

Ad-hoc

Low power App

App3
App2
App1

Supervisor

Linux O/S
Device management is not an option: Every processor and every application need to be configured, upgraded and monitored.
Each protocol must be secured.

And synchronized:
You can’t trigger an update with a protocol, while you are rebooting the device using another.
M2M/IoT is not a simple problem. Security and provisioning are really the hardest ones. Try hard to reduce the number of protocols to make your life easier!
ONE PROTOCOL?

Time to choose :o)
THERE IS NO PROTOCOL TO RULE THEM ALL?

Every protocol is nice and specialized?
ONE PROTOCOL?

CoAP with LWM2M provides efficient device management and application protocol.

It can be the only protocol of your device to rule them all!
THANKS!

Questions?

More Questions?
@vrmvrm
jvermillard@sierrawireless.com