7th International
LED professional Symposium +Expo
Sept 26-28, 2017 | Bregenz

How the OpenAIS Group Communication allows secure and low latency interoperable IoT based Lighting Controls Designs

Borsoi G., Werner W.
TRIDONIC, OpenAIS

Table of Content

- IoT-based lighting controls
  - Constrained
- Benefits & challenges
  - Scalability
  - Features development
  - Interoperability(?)
  - Latency
  - Synchronization
  - Security
- Solutions
  - CBOR
  - Peer-to-peer communication
  - Group communication
  - OpenAIS secure peer-to-peer group communication
- Lessons learned

IoT-based lighting controls - I

- **Internet** of Things:
  - Well tested network architecture
  - Supporting
    - a wide variety of services
    - a wide range of requirements
  - Normally:
    - high bandwidth
    - low latency
    - connection-oriented
  - Secure(able):
    - Online banking
    - Certified e-mails
  - Possibly a bit complicated from time to time

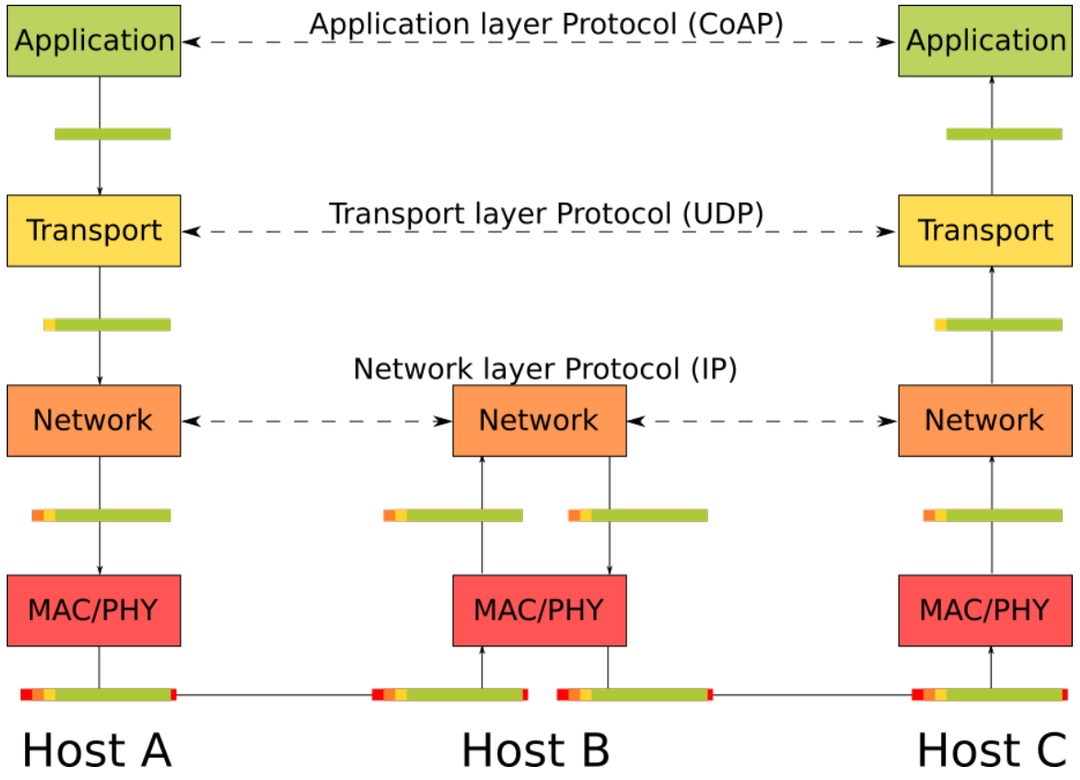
IoT-based lighting controls - II

- Internet of **Things**:
  - Billions of nodes
  - Cheap nodes
  - Resource-constrained nodes

- MCUs and SoCs:
  - <1MB flash
  - <128kB RAM
  - <100MHz Cortex M
  - Relatively powerful compared to those needed in LED drivers…
  - ...but still many orders of magnitude smaller than any conventional IP host.

Benefits

- The benefits mainly come from the Internet Protocol Suite
  - Ideally well layered architecture
  - Ideally independent layers
  - Non-proprietary
  - Lack of central administration
  - 128bit addresses
  - Security



Benefits - Scalability

- Unlimited number of nodes
  - That is,128bit addresses;
- Integration in any (suitable) IP infrastructure
  - That is, provided IP is used and the PHY is the same across all nodes;
- Integration in any IP infrastructure with suitable gateways
  - That is, to connect different networks (e.g. wired and wireless);
- No Application Gateways –> no translations
  - That is, in addition to the translation from user input and to actuator output;

Benefit – Feature development

- New features/functionality needs to be added only at the end nodes;
- The infrastructure stays the same
- Not necessarily your own features…

Chance - Interoperability

- Most likely
  - There are some alliances that try to achieve this (e.g. OpenAIS);
- For sure at the Network layer
  - A shared infrastructure is definitely already possible;
- Not limited to lighting applications
  - But lighting is everywhere so it is a good candidate for (low power wireless) infrastructure;

Challenges

- The challenges mainly come from the Resource Constraints:
  - Small MCUs with limited computing power -> delays
  - Limited availability of RAM -> needs efficient implementation, limits network size
  - Relatively complex tasks -> suggests use of OS, rather than BM

- But not only:
  - History
    - Routing schemes are discriminated (Unicast vs. Multicast)
    - Transport protocols are not the same (TCP vs UDP)
  - Numbers – lots of devices to handle, commission and control
  - Public routing and fixed IP address – Alice, Bob, Eve…

Challenges – Latency

- At the node:
  - Limited computing power;
  - Encryption/decryption;
  - Relatively big packets;
- On the way:
  - Limited bandwidth;
  - Shared medium;
  - Routing;

Challenges – Synchronization

- As in: “getting Things to do stuff at the same time”

- Connections:
  - TCP requires a connection to be established and additional overhead;
  - UDP is much more lightweight and supports additional…

- Routing schemes:
  - Traditionally Unicast is better supported;
  - More recently applications requiring Multicast emerged/became feasible (e.g. live video streaming, online gaming…);
  - Lower network traffic;

Challenges – Security

- Implementation in constrained systems is challenging
  - Delays – computing time is relevant and proportional to security;
  - Key exchange – must be done properly and requires some tools;
  - Storage – encryption material takes up lots of valuable storage;
  - Updates – critical updates to fix security gaps are delicate;
- A good balance between security and performance is needed

OpenAIS Group Communication (OGC)

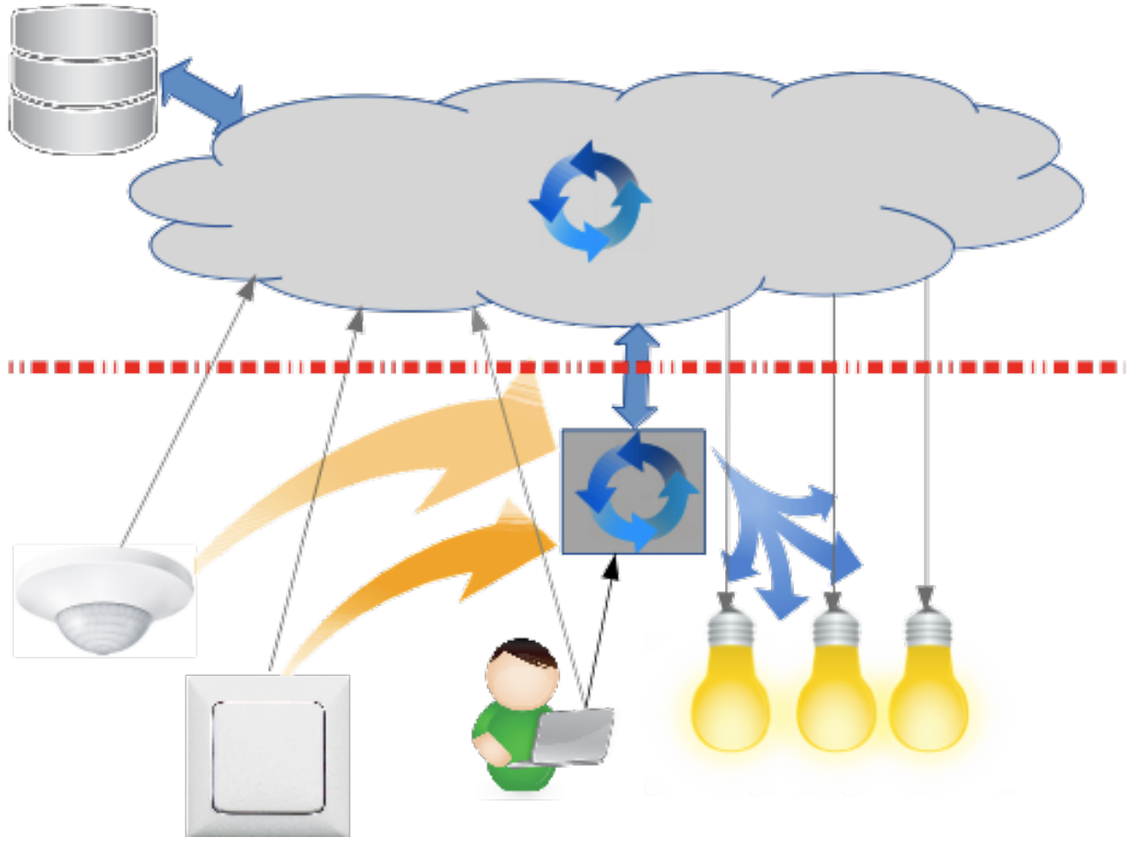
- Group communication (multicast)
  - By setting-up and using suitable IPv6 Multicast addresses;
  - UDP of course;
- Direct (Peer-to-peer) communication
  - By having part of the controller logic at the end-nodes;
- Security
  - By using encryption at the Object level for multicast messages;

OGC – decentralized controls

- Direct (P2P) communication:
  - Improves robustness against failures of centralized controller;
  - Improves robustness against infrastructure network congestion;
  - minimizes routing delays;
  - Reduces latency;

- Additional layer optimized for low latency

- Setup with IoT Framework.

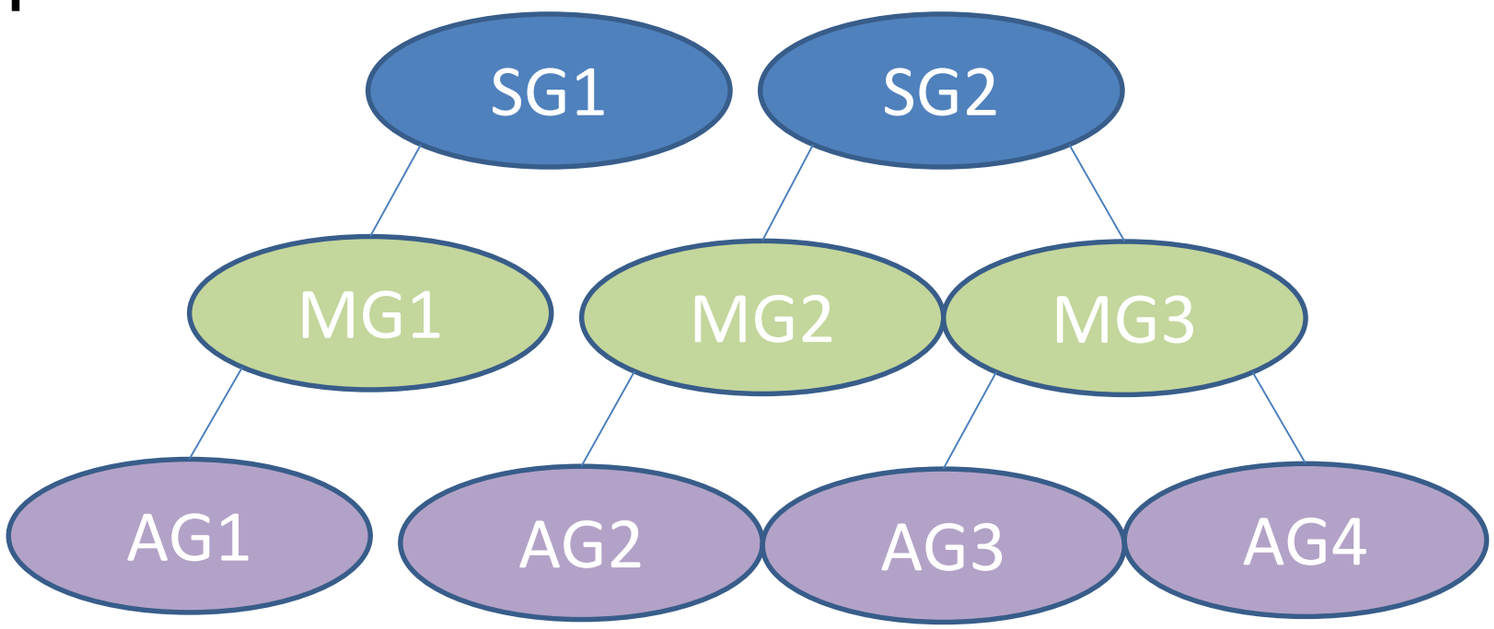


OGC – Security I

- OA uses
  - asymmetric keys for unicast communication;
    - more secure
    - more storage needed
    - provides Authentication (ECDH)
    - DTLS
  - symmetric keys for OGC;
    - faster
    - smaller
    - less secure
    - symmetric keys are generated by a KDC
    - COSE/OSCOAP (CBOR Object Signing & Encryption/Object Security of CoAP)

OGC – Security II

- OA defines 6 Authorization levels:
  - ☑ Level 0: Object detection
  - ☑ Level 1: Reporting only
  - ☑ Level 2: Standard use
  - ☐ Level 3: Commissioning use / parameterization services
  - ☐ Level 4: Commissioning use / localization and addressing services
  - ☐ Level 5: Device Owner
- OA defines 3 group types:
  - Application Groups (AG)
  - Multicast Groups (MG)
  - Security Groups (SG)



Lessons Learned

- Challenges in IoT lighting control implementation
  - Constrained devices are powerful enough for the IoT
- OpenAIS specified viable solutions for such challenges
  - Using a mix of standard technologies appropriately
- OGC addresses the needs of lighting installations:
  - Limiting latency using Multicast messaging
  - Limiting encryption/decryption overhead for time-critical communication by using symmetric key cryptography
  - Allowing cryptographic material to be shared among multicast groups and application groups
- OGC allows the design of interoperable systems without limiting product differentiation