WEBINAR

Designing a Secure Cloud-Enabled Peer-to-Peer IoT Application

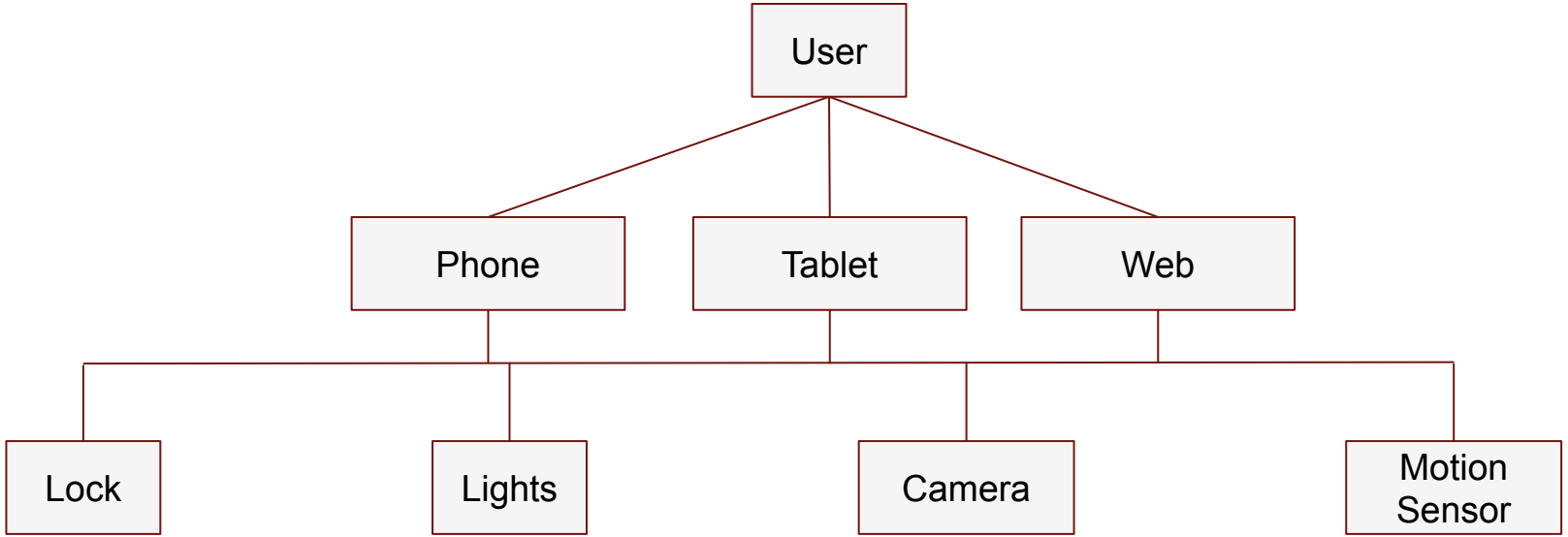
July 9, 2021

AGENDA

1. Application Requirements and Preliminary Architecture
2. Design Alternatives
3. Solution Overview
4. OpenDDS Features Behind the Solution
5. Lessons Learned and Conclusions

SAMPLE APPLICATION - HOME AUTOMATION AND SECURITY

Imagine you are a manufacturer of home automation and security devices



REQUIREMENTS - USER

Low latency for a good user experience

- Lock/light should change when the user presses the button

Remote access

- Users must be able to check and control their homes remotely

Secure

- Only authorized users should be able to access and control the home

Evolvable

- Many different device and application versions in play at any given time

REQUIREMENTS - OPERATIONS

Device administration

- Support firmware upgrades and log collection

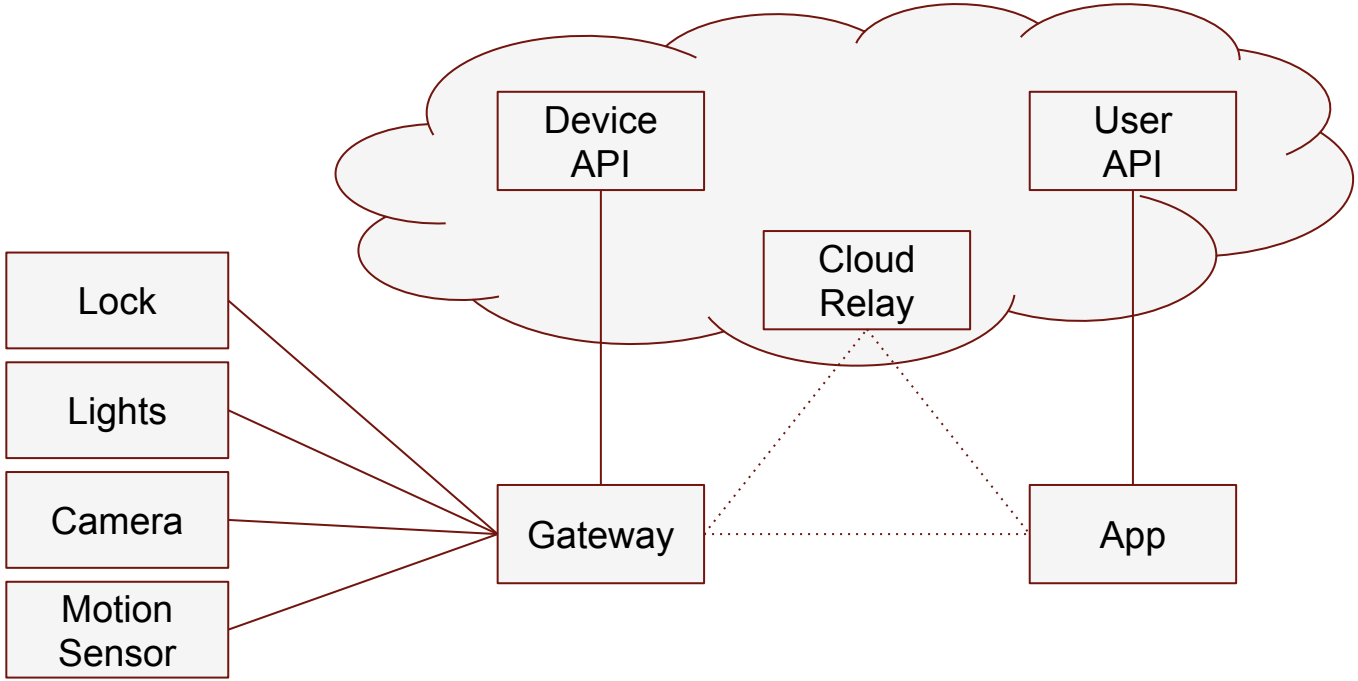
Reduce cloud costs

- Maximize profitability if offered as a paid service
- Minimize cost if offered for free

Common technology stack

- Reduce overhead associated with adding “yet another technology”

PRELIMINARY ARCHITECTURE



DESIGN ALTERNATIVES - PROBLEM STATEMENT

- Gateway is a device installed in the home
  - IP connectivity to the Internet
  - Probably behind NAT/firewall
- App is a mobile application running on a phone or tablet
  - May be on the same network as Gateway (if user is at home)
  - May be taken outside the home, connected to WiFi or mobile data
- How should these two components communicate with each other?
  - Send messages through the cloud
  - Exchange messages directly
  - Some combination of these
- Also: Can a cloud-hosted application communicate with the Gateway?

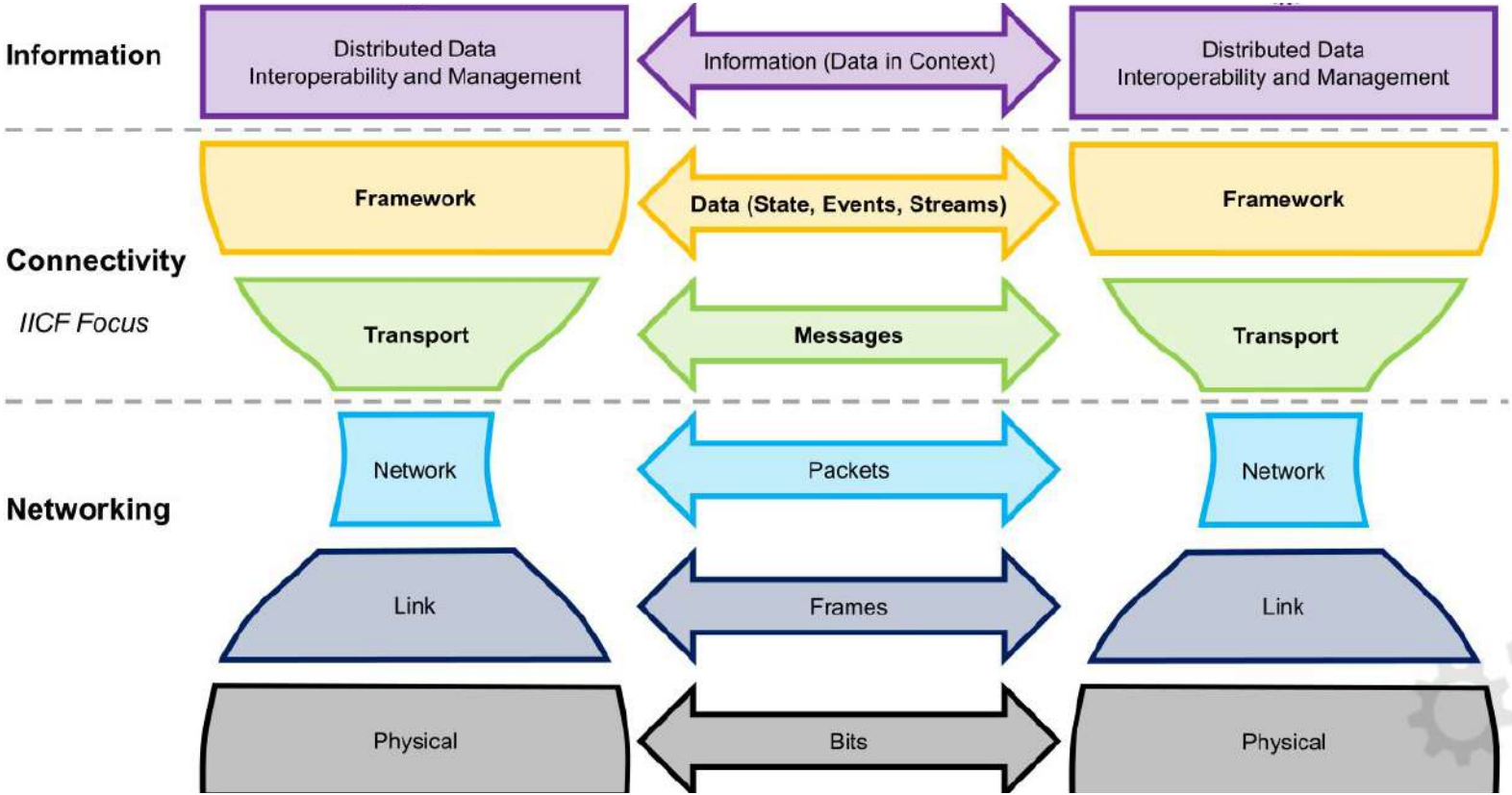
DESIGN ALTERNATIVES - BREAKDOWN

- Discovery: How do the two components find each other?
  - Local: mDNS, ZeroConf/Bonjour, UPnP
  - Cloud: REST-based services at well known location
- Security: Is this App (user) authorized to communicate with this Gateway?
  - Does the Gateway itself make this decision based on how it's provisioned/registered?
  - Does a cloud-hosted service mediate access to the Gateway?
- Commands: How does the App request changes on the Gateway?
  - Plain HTTPS or use a messaging system like MQTT, CoAP, AMQP
- Notification: How are events that occur on the Gateway relayed to the App?
  - HTTPS polling, WebSocket (including wrappers like SignalR)
  - Messaging system (see above)

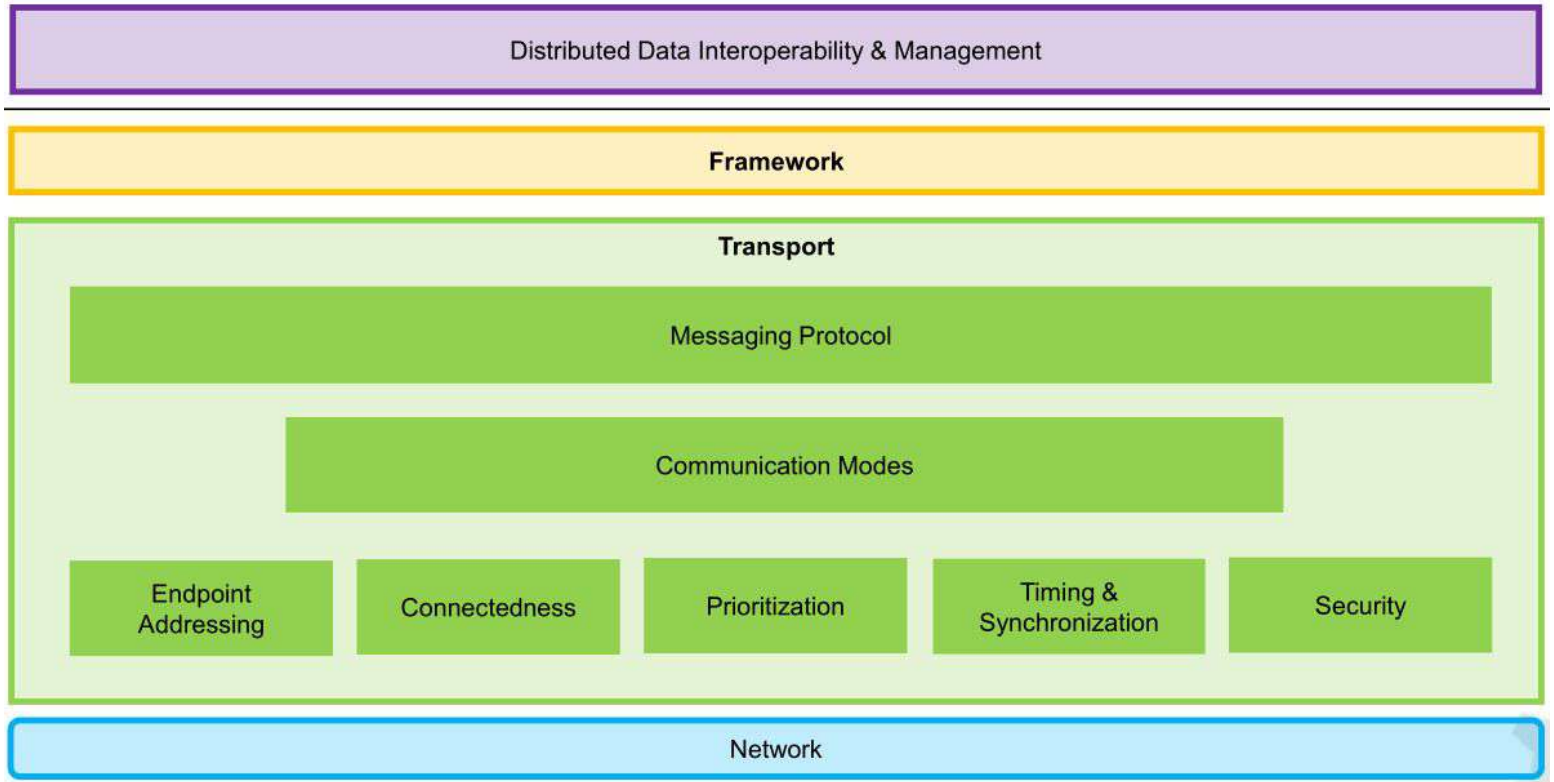
DESIGN ALTERNATIVES - EVALUATION CRITERIA

- Generality
  - Does the same protocol work for both local and remote? Are different QoS supported?
- Infrastructure Needs
  - Are brokers required? If so, how is High Availability / Fault-Tolerance handled?
- Resilience to Network Issues
  - Does the solution support connecting the authorized App to the Gateway even during an internet service outage?
- Efficiency
  - Latency, overall number of bytes exchanged, cloud ingress/egress, mobile battery drain
- Abstraction
  - Does the application developer need to be concerned with byte-by-byte encoding or just high-level types?

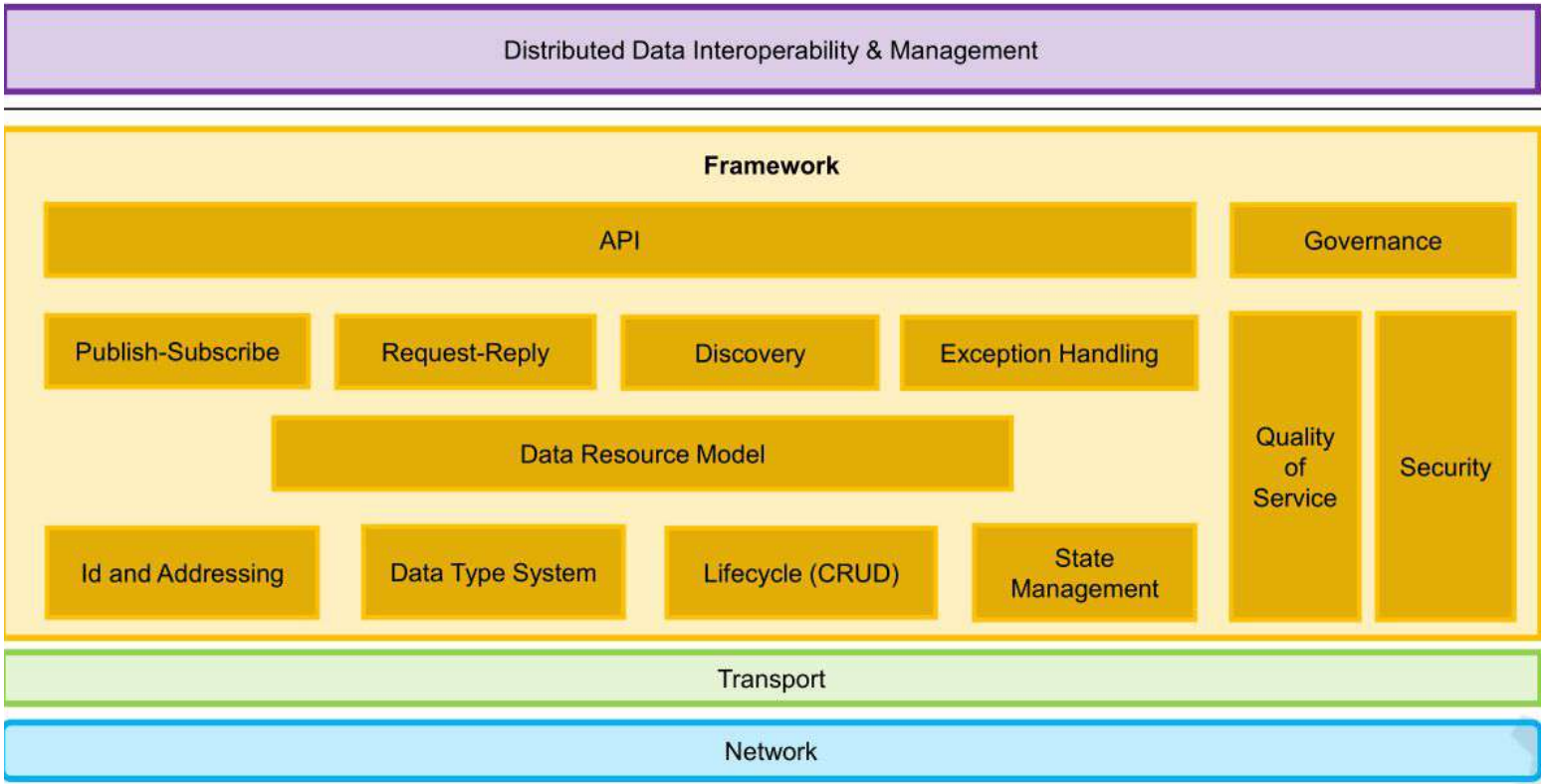
IIC CONNECTIVITY FRAMEWORK



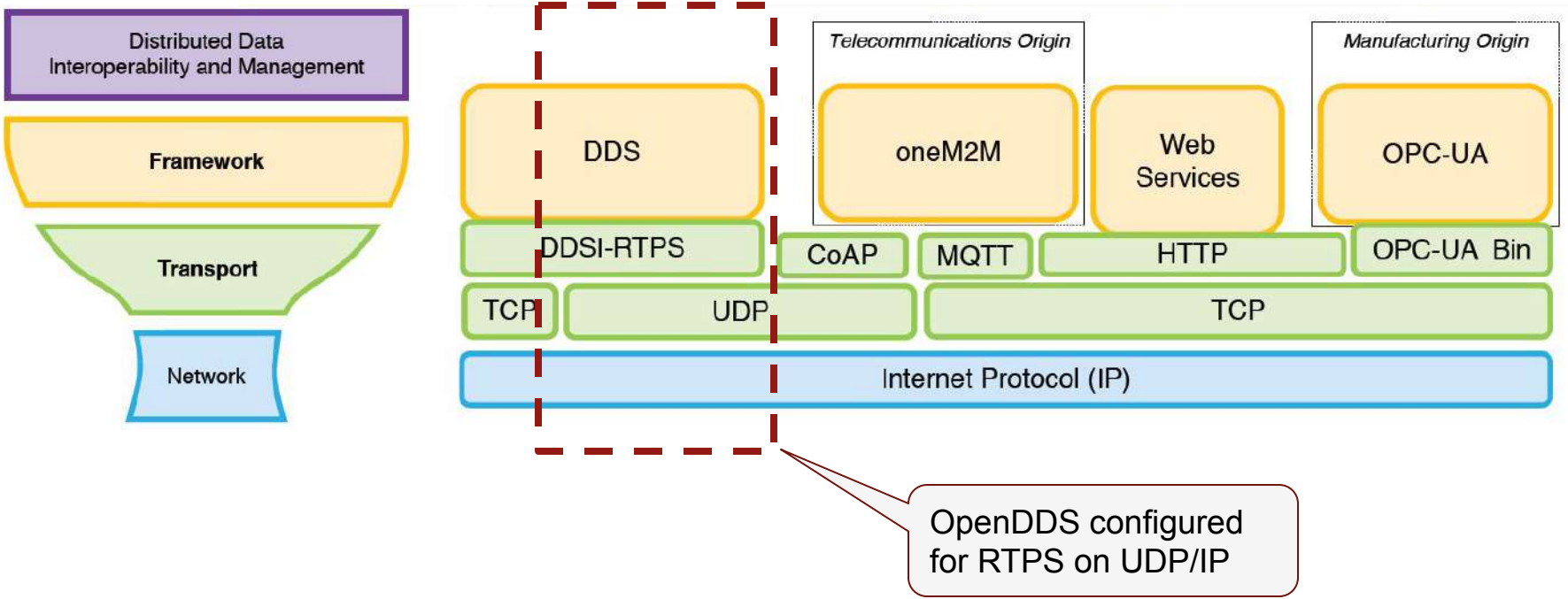
IIC CONNECTIVITY FRAMEWORK



IIC CONNECTIVITY FRAMEWORK

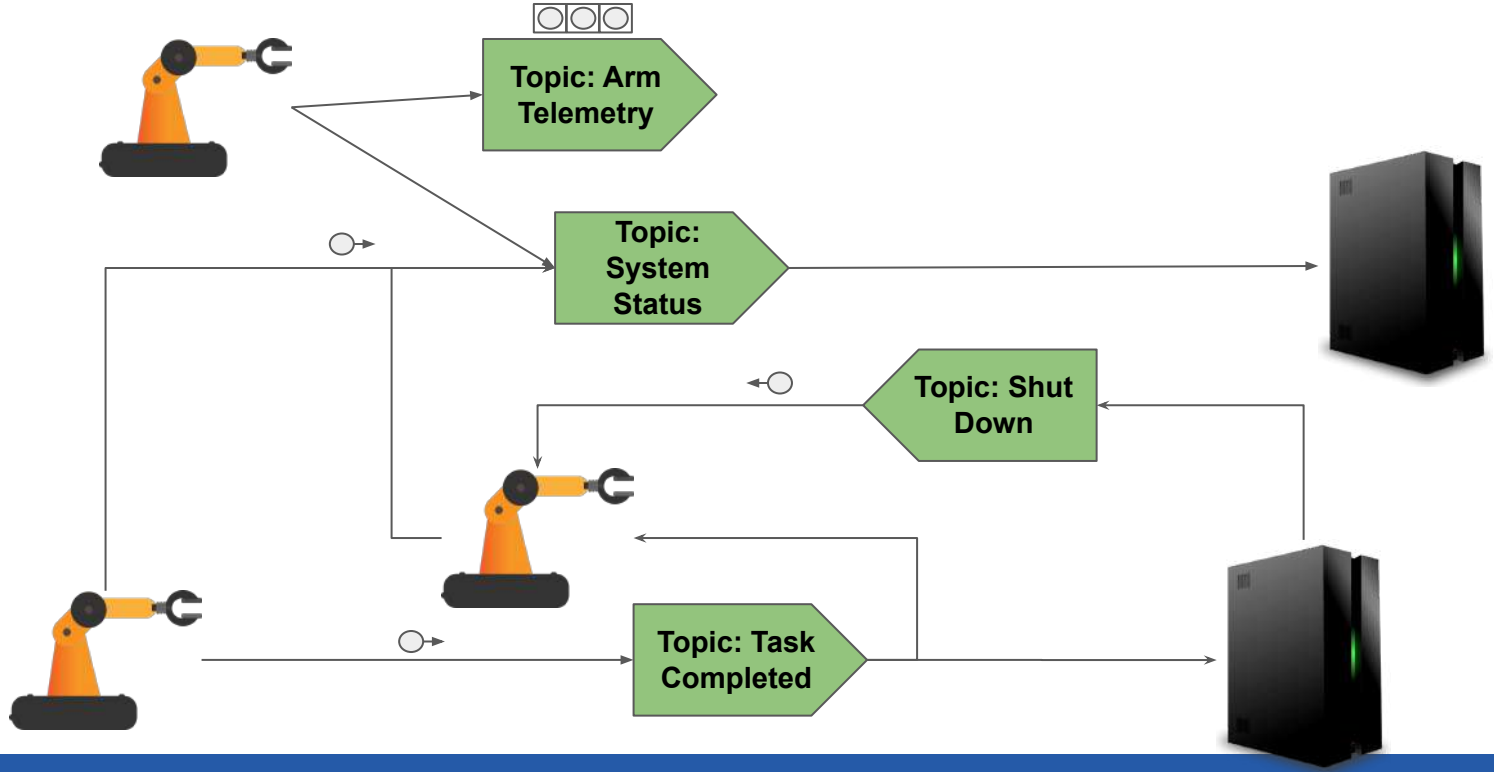


IIC CONNECTIVITY FRAMEWORK



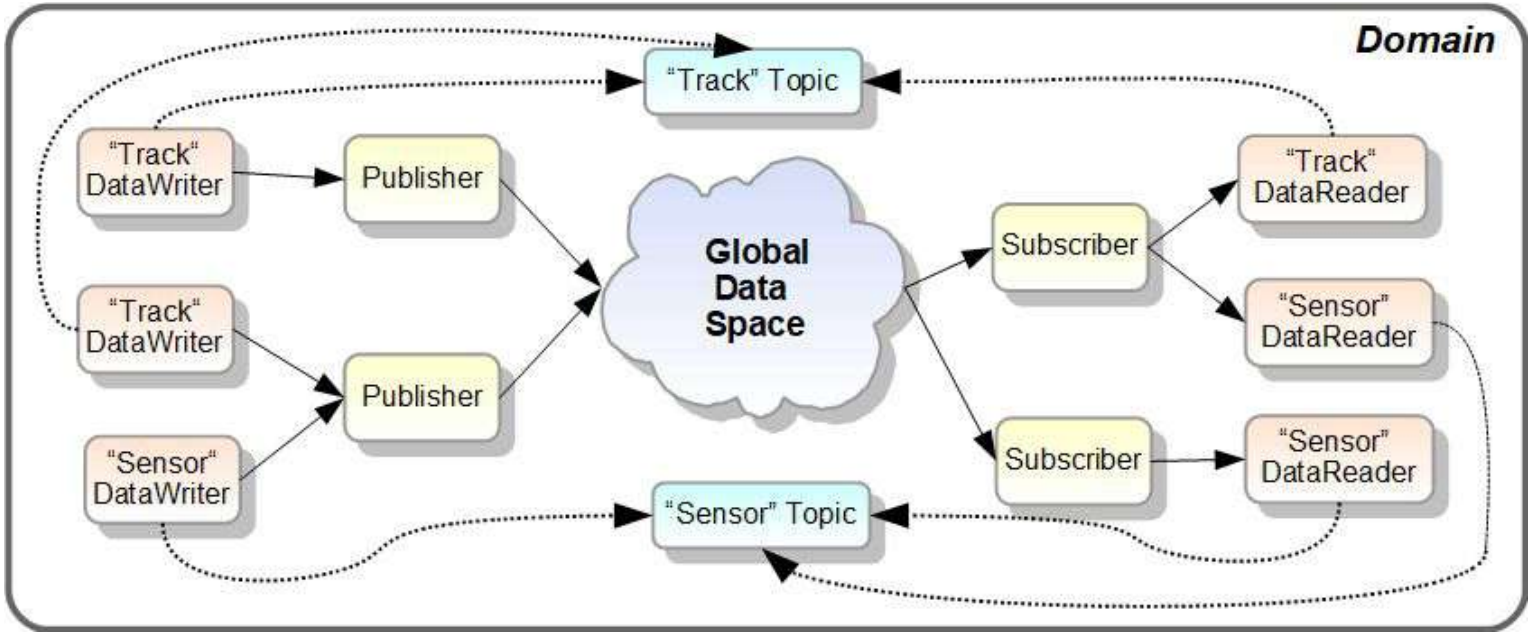
BRIEF INTRODUCTION TO OpenDDS

OpenDDS applications efficiently share data across the network using strongly-typed and asynchronous cache updates based on Topics and QoS policies.

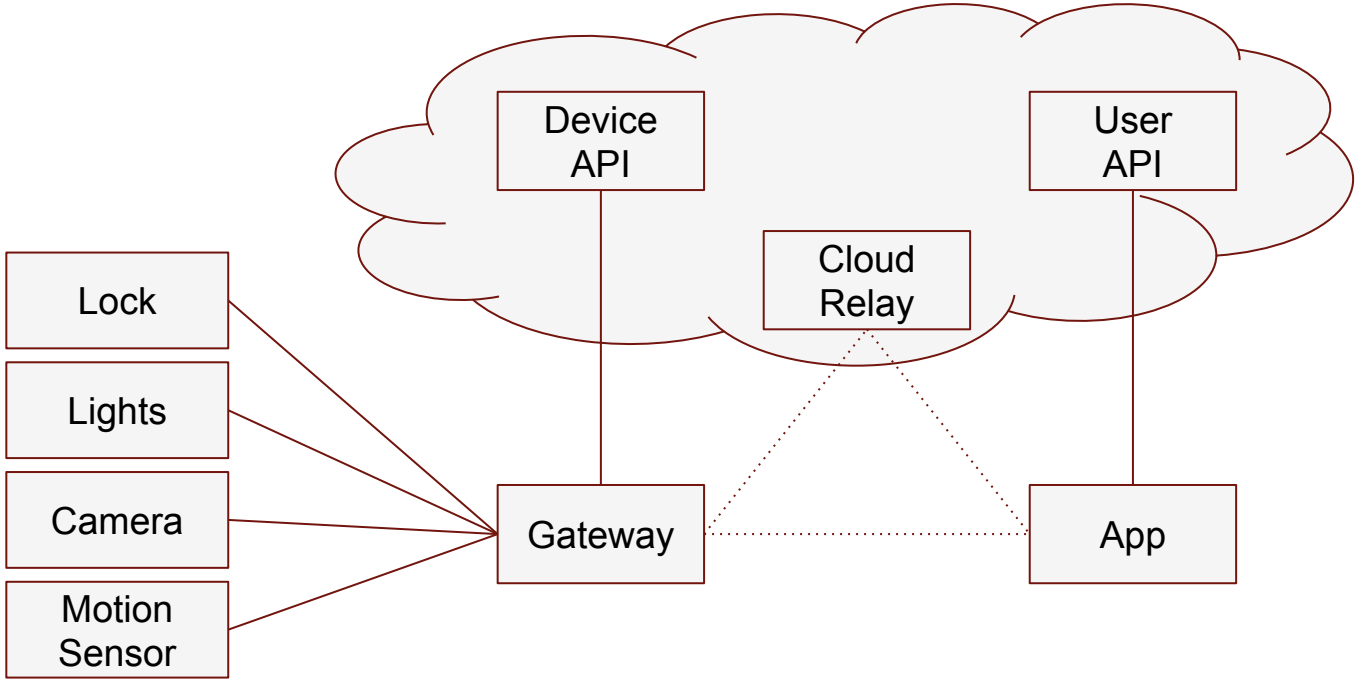


BRIEF INTRODUCTION TO OpenDDS

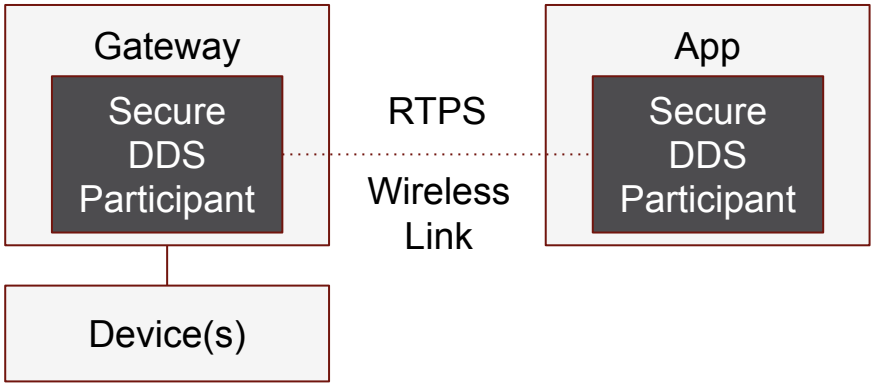
- Sender of data: DataWriter
- Receiver of data: DataReader
- Coordination point: Topic
- Overall data space: Domain
- Access to data space within an application: DomainParticipant



RECALL THE PRELIMINARY ARCHITECTURE



SOLUTION OVERVIEW - LOCAL INTERACTIONS



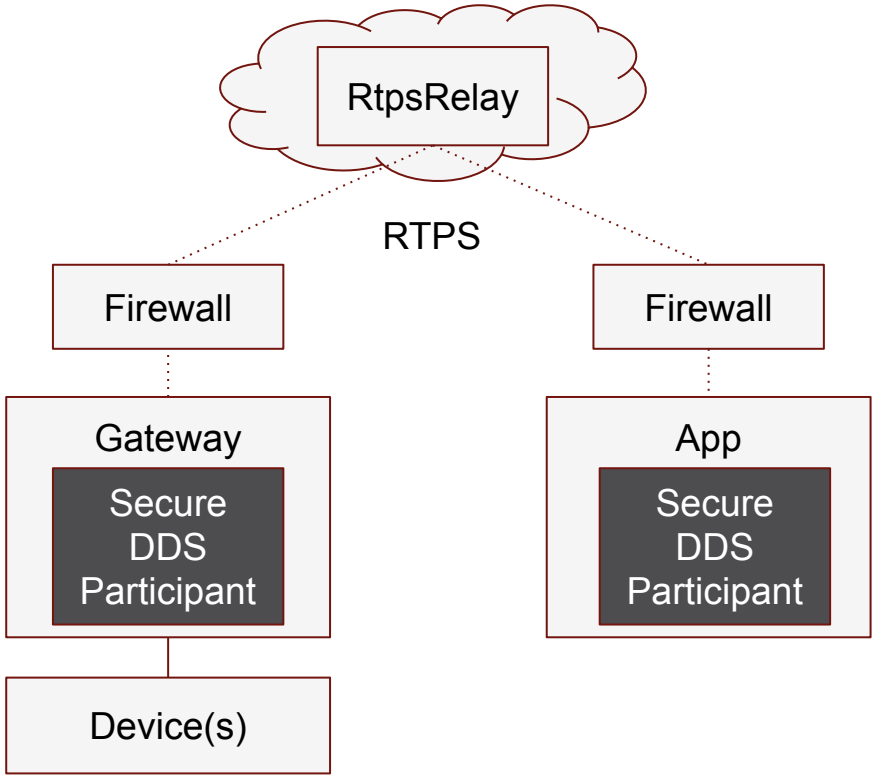
App and Gateway discover each other using multicast

They then authenticate, associate, and exchange data

DDS Security for authentication and authorization

RTPS (Real-Time Publish/Subscribe) is an interoperable wire protocol for DDS that supports DDS Security

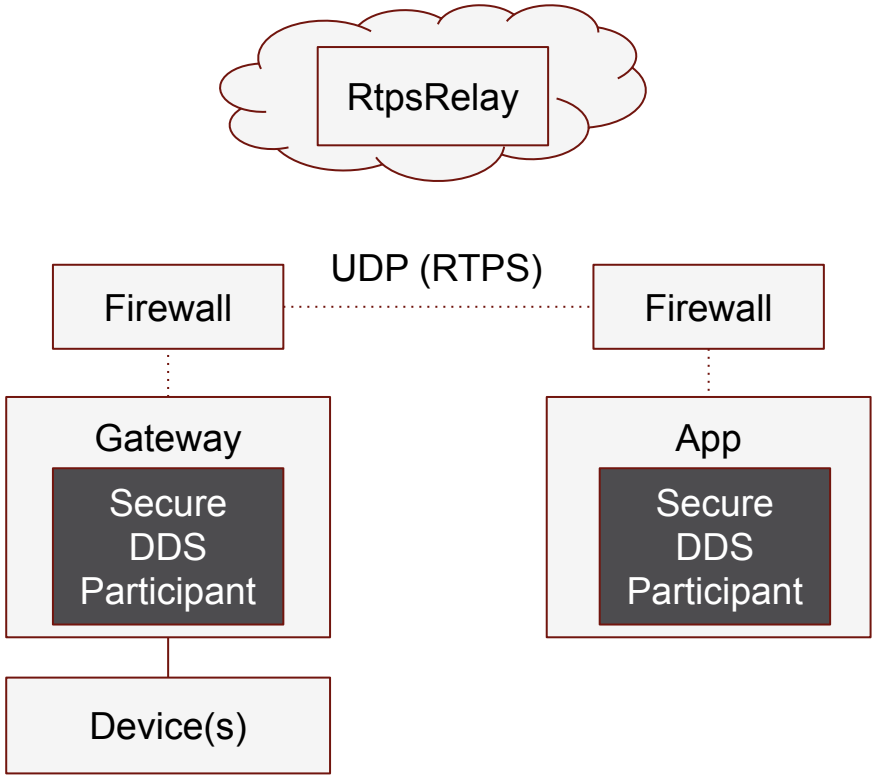
SOLUTION OVERVIEW - REMOTE INTERACTIONS



RtpsRelay

- Generic
- Forwards RTPS packets to appropriate participant
- Horizontally scalable
- Supports DDS Security

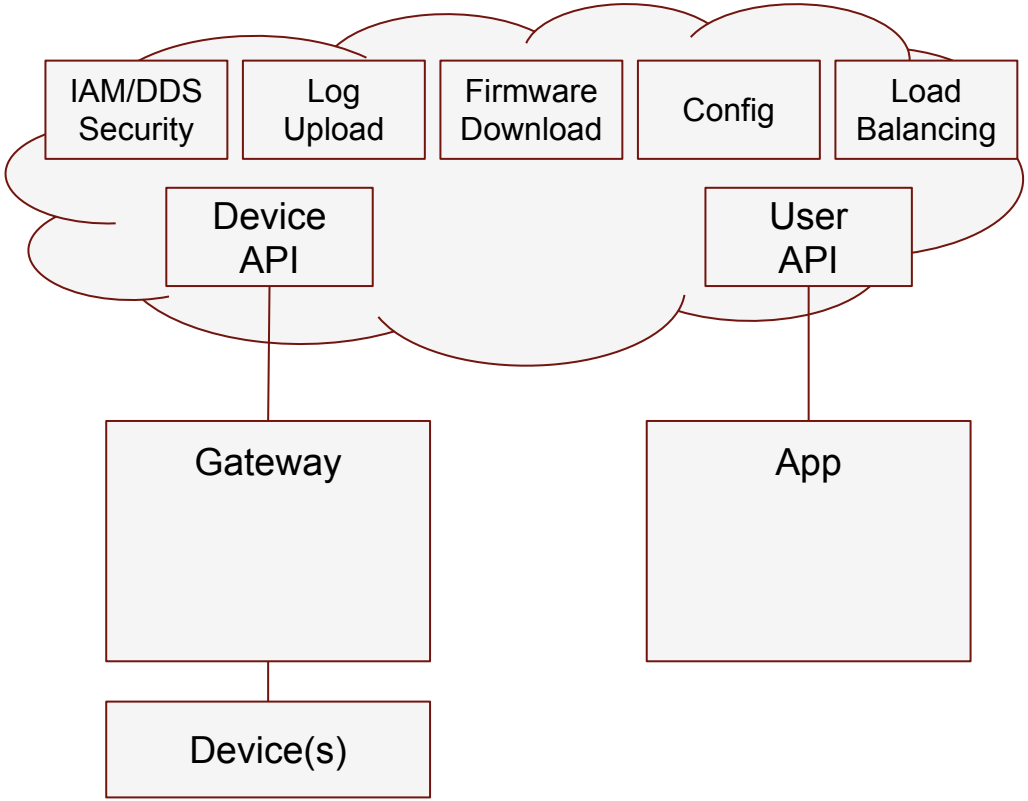
SOLUTION OVERVIEW - REMOTE INTERACTIONS WITH ICE



Interactive Connectivity Establishment (ICE)

- Allows two participants that are behind firewalls to exchange messages
- Requires a 3rd party (RtpsRelay) for exchange of discovery information

SOLUTION OVERVIEW - REST APIs

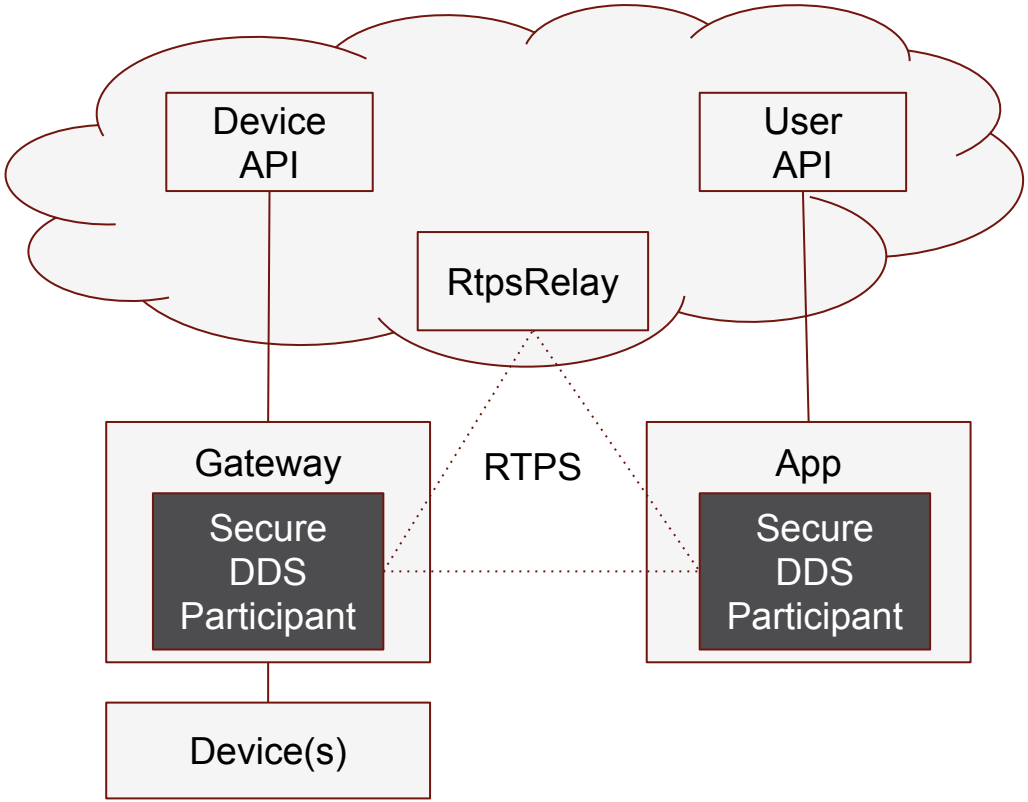


Services provided by REST APIs

- Identity and Access Management (IAM)
  - Translation to DDS Security
- Log Upload*
- Firmware Download*
- Configuration*
- RtpsRelay load balancing

* are standard APIs. Use the technology that is right for you!

SOLUTION OVERVIEW

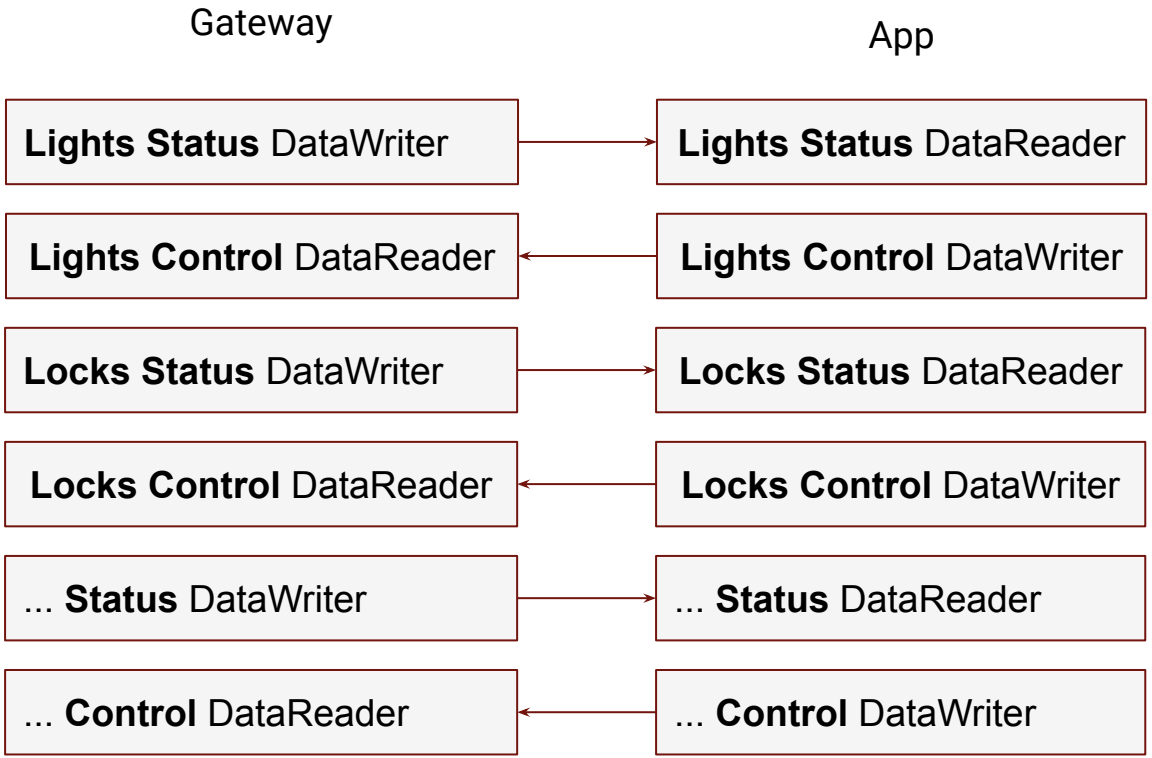


The local interaction is driving the solution

Avoid the cloud => decrease latency, minimize cloud spend

Use the same technology for local and remote => common technology stack

SOLUTION DETAIL - GATEWAY AND APP



Status topics (durable, reliable) are written by the Gateway and read by the App. (DDS has a variety of QoS policies.)

Durability causes App to receive status upon connection.

Control topics (reliable) are written by the App and read by the Gateway.

SOLUTION DETAIL - EVOLVABILITY USING XTypes (eXtensible Types)

What happens when the definition of a topic changes?

```
@topic
struct LightControl {
  boolean state;
};
```

```
@topic
struct LightControl {
  boolean state;
  // What will they think of next?
  float level;
  float temperature;
};
```

XTypes allows one to plan for changes in the structure of topics.

```
@topic
@mutable
@autoid(HASH)
struct LightControl {
  boolean state;
};
```

Readers can supply defaults for missing fields and ignore extra fields.

SOLUTION DETAIL - DDS SECURITY

Authentication

- Each participant has a private key and public key (certificate)
- Issued by a mutually trusted Certificate Authority (Identity CA)

Authorization

- Common governance file sets policy for RTPS and topics
- Each participant has a permissions file that describes how they can interact with the domain
- Signed by a mutually trusted Certificate Authority (Permissions CA)

SOLUTION DETAIL - DDS SECURITY GOVERNANCE FILE

```
<dds>
  <domain_access_rules>
    <domain_rule>
      <domains>
        <id>0</>
      </domains>
      <allow_unauthenticated_participants>FALSE</>
      <enable_join_access_control>TRUE</>
      <discovery_protection_kind>NONE</>
      <liveliness_protection_kind>NONE</>
      <rtps_protection_kind>ENCRYPT</>
      <topic_access_rules>
        <topic_rule>
          <topic_expression>*</>
          <enable_discovery_protection>FALSE</>
          <enable_liveliness_protection>FALSE</>
          <enable_read_access_control>TRUE</>
          <enable_write_access_control>TRUE</>
          <metadata_protection_kind>NONE</>
          <data_protection_kind>NONE</>
        </topic_rule>
      </topic_access_rules>
    </domain_rule>
  </domain_access_rules>
</dds>
```

Governance file answers

- Is authentication required?
- Are RTPS messages plain, signed, or encrypted?
- For topics matching an expression
  - Should their liveliness/discovery messages use secure endpoints?
  - Does a writer/reader need explicit permission to write/read the topic?
  - Are RTPS submessages and data payloads plain, signed, or encrypted?

SOLUTION DETAIL - DDS SECURITY PERMISSIONS FILE

```
<dds>
  <permissions>
    <grant name="HomeAutomation">
      <subject_name>/CN=device1</>
      <validity>
        <not_before>2021-06-23T03:08:37.288Z</>
        <not_after>2021-07-08T03:08:37.288Z</>
      </validity>
      <allow_rule>
        <domains><id>0</></>
        <publish>
          <topics><topic>Light Status</></>
          <partitions><partition>home1</></>
        </publish>
        <subscribe>...</>
      </allow_rule>
      <default>DENY</>
    </grant>
  </permissions>
</dds>
```

Permissions file answers

- To whom do the permissions apply?
- When are permissions valid?
- On which topics and partitions can this participant publish/subscribe?
- What is the default policy?

Same function as a JSON Web Token (JWT)

SOLUTION DETAIL - DDS SECURITY INFRASTRUCTURE

A (web)server that provides the following capabilities:

- Provides CA certificates
- Provides governance file
- Issues certificates to participants
- Issues permission files to participants

- Update a CA when it expires
- Update a governance file if necessary (Yes, we did this in a live system.)
- Issue new certificates when they expire
- Issue new permissions when they *change* or *expire*

SOLUTION DETAIL - DDS SECURITY INFRASTRUCTURE

For authentication, DDS security can use a Certificate Revocation List (CRL) or the Online Certificate Status Protocol (OCSP).

Industry seems to be moving away from these in favor of certificates with short lifespans.

For DDS Security, you can do any of the following:

1. Use a CRL or OCSP [neither currently implemented in OpenDDS].
2. Issue certificates with short lifespans.
3. Issue permissions with short lifespans.
   (Caveat: A participant may be able to discover other participants but it won't be able to interact with them.)

SOLUTION DETAIL - DDS SECURITY INFRASTRUCTURE

Somewhere in the system, there is a database that describes the relationship between users and their gateway(s). This data is used to generate permissions files.

Alice	Whiteacre
Bob	Blackacre
Eve	Greenacre
Eve	Brownacre

```
…
<subject_name>/CN=Eve</>
…
    <partitions>
      <partition>Greenacre</>
      <partition>Brownacre</>
    </>
…
```

SOLUTION DETAIL - DDS SECURITY INFRASTRUCTURE

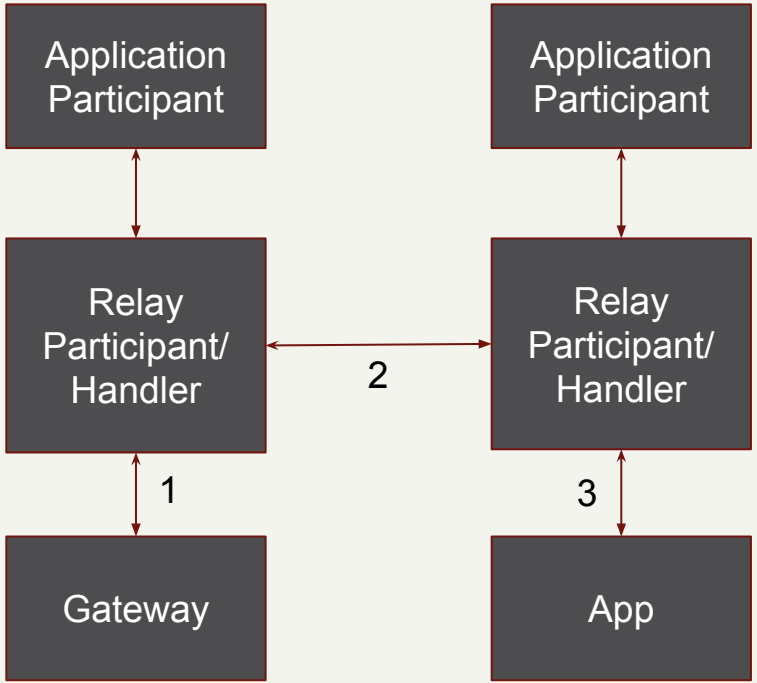
Permissions change

- Limiting validity of permissions prevents a participant from getting too far out of date.
- Someone with physical access to the gateway should be able to restrict access to it.

DDS Security Infrastructure could be primary or secondary

- Primary if participants only have DDS Security documents
  - Could authenticate to web APIs using client certificate
  - Challenges if certificate expires
- Secondary if system contains a distinct IAM solution
  - Credentials can be used to retrieve DDS Security documents
  - DDS Security documents are disposable

SOLUTION DETAIL - RtpsRelay



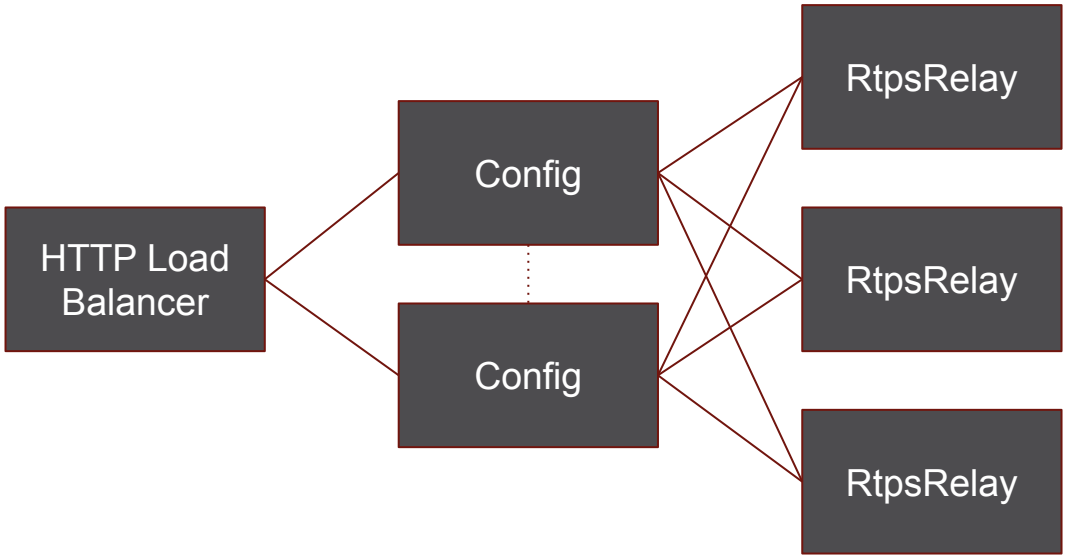
Application participant discovers all connected participants so it can learning a routing table.

Routing table is shared to other relay instances via the Relay Participant (DDS back-channel).

1. Gateway sends message to relay.
2. Relay forwards to peer.
3. Peer forwards to App.

SOLUTION DETAIL - RtpsRelay AND LOAD BALANCING

Add a (HTTP) Config server so clients can find an RtpsRelay instance. RtpsRelay instances publish statistics that Config server can use for load balancing decision.



SOLUTION DETAIL - RtpsRelay Cost

Observations

1. Gateways are idle 99% of the time.
2. Some form of polling is unavoidable (Gateways check connectivity to relay).
3. Pay for messages sent by relay.

Conclusion

1. Avoid the relay when possible => Local, ICE.
2. Response to polling request should be small => RtpsRelay uses STUN response (96 bytes).
3. Cost model is idle gateway + active user.

LESSONS LEARNED

- Pick a technology that makes sense
- You can't achieve scalability by accident
  - Design, model, and test
- Plan to have a server for DDS Security (that is tied to your IAM system)
- Integrate the RtpsRelay load balancer with the IAM system for flexibility and traceability
- The cloud doesn't support multicast (convert multicast to unicast for Relay Participant)
- You will face IP fragmentation issues across the Internet and in the cloud
- RTPS Discovery with security can be slow and intensive
  - HTTP may be better for a one-off update from a mobile device

CONCLUSIONS

Webservices are useful for device administration activities

Peer-to-peer interactions can reduce latency and reduce cloud costs

Enabling peer-to-peer interactions in an IoT application requires solutions for discovery, security, and evolvability

- OpenDDS addresses these with RTPS discovery, DDS Security, and XTypes

A fall-back to the cloud is necessary

- OpenDDS addresses this with the RtpsRelay

Common technology stack vs. different solutions for local and cloud

DDS Security Infrastructure will be part of the application's IAM

FOR MORE INFORMATION

Industrial Internet Consortium https://www.iiconsortium.org/IICF.htm

OMG DDS Foundation https://www.dds-foundation.org

OpenDDS https://opendds.org

Object Computing, Inc. https://objectcomputing.com