Agenda

- Eclipse IoT Working Group Strategy for 2018 Review
  - Overview/Context and Purpose
  - Opt-in Community Norms
  - Steps Toward Coordinated Releases and Integrated Stacks
  - Communication Between Projects

- Hono, Kapua, Kura - Messaging Integration
  - IoT Messaging Interactions
  - MQTT Topic Namespaces and AMQP Addresses

- Hono, Kapua - Service Integration
  - Kapua Services as Hono Providers

IoT Messaging Interactions

Devices - Devices can connect to the Cloud and have an identifier.
Devices can be smart devices (gateways) or simple constrained devices.
AI: Picture and Glossary

Interaction Type	Description	Notes
Device-to-Cloud (D2C)	Unsolicited messages sent by devices to the remote application (including Kapua). The conversation can be send-and-forget or request-reply.	
Cloud-to-Device (C2D)	The remote application initiates a conversation with a target device. The conversation can be a send-and-forget or request-reply.	
Device-to-Device (D2D)	The device initiates a conversation with another target device. The messages are initiated from a device and targeted to another device; messages will follow a device-to-cloud-to-device. The conversation can be a send-and-forget or request-reply.	
Platform-to-Cloud (P2C)	Message produced by the messaging platform to the application. E.g. Connection monitoring and life-cycle messages.	

Discussion resolution: do not focus D2D.

Device-to-Cloud (D2C) Messaging

Unsolicited Messages Concepts

Message Type	**Description**	**QoS Assumption**	**Volume Assumption**	**Priority Assumption**	**Persistence Assumption**
Telemetry	A raw data reading - e.g. the value of a modbus register.	Low (At most once) High (At least once)	High	Low	No
Events	A data reading has reached a threshold and that triggered an event. A new configuration has been detected or applied.	High (At least once)	Medium	High	Yes
Alerts	Something happened and the device may not operate properly	High (At least once)	Low	Very High	Yes

Decision:

- QoS exactly once will not be supported

Berlin Namespace - Unified Kura, Kapua Hono Messaging Namespace

Eclipse IoT Namespace Proposal

Message Type	Direc-tion	MQTT 3.1.1 Topic	MQTT QoS	MQTT Payload	Hono AMQP Address	Payload	Hono Message Persistence
Telemetry	D2C	t/tenant/device/	0, 1	Opaque	telemetry/tenant/	Opaque	No
Events	D2C	e/tenant/device/	1	Opaque	event/tenant/	Opaque	Yes
Alerts	D2C	a/tenant/device/	1	Opaque	alert/tenant/	Opaque	Yes
Life-Cycle	P2C	-	-	-	event/tenant/ ~~platform/tenant~~	Defined by Hono	Yes, by conf.
Device Subscription	C2D	c/tenant/device/#	0		-	Hono-TODO define how MQTT subscriptions are handled	-
Request Topics	C2D	c/tenant/device/corrID	0	Opaque	control/tenant/device	CorrID in header	No
Response Topics	D2C	r/tenant/device/corrID	0	Opaque	control-reply/tenant/device	CorrID in header	No
Device to Device	D2D	While possible with the MQTT namespace above, we may not want to enable it for security reasons.					

Notes:

- Can the Messaging Infrastructure add device-specific message headers into the messages? For example, data vs control message?
- Possible examples are: originating protocol, content-type per message type, ...

Service Integration - Device Registry

Tenant API

API	AMQP 1.0 Endpoint	Response	Notes
Get Tenant Registration	tenant/,get,tenantId	Tenant status and configuration for protocol adapters.	Post authentication

Device API

API	AMQP 1.0 Endpoint	Response	Notes
Assert Device Registration	registration/tenant,assert,deviceId[,gatewayid]	Signed JSON Web Token with device's tenant and validity period. Can also return default properties of the device.	Post authentication and connection
Get Registration Information	registration/tenant,get,deviceId		

Device Model

- Tenant Id
- Device Id
- Optional name/value pair default properties

Base Implementation

- https://github.com/eclipse/hono/tree/master/service-base
- https://github.com/eclipse/hono/tree/master/service-base/src/main/java/org/eclipse/hono/service/registration

Service Integration - Credentials (D2C)

Credentials API - Used in the Device-to-Cloud (D2C)

API	AMQP 1.0 Endpoint	Response	Notes

Get Credentials	credentials/tenant/,get,”hashed-password”,auth-id[,client-id]
	Hono-TODO: add connection property bag with “client-id” as a well-known property in that set.
		`{`
		`"device-id": "4711",`
		`"type": "hashed-password",`
		`"auth-id": "sensor1",`
		`"enabled": true,`
		`"secrets": [{`
		`"not-after": "2017-12-24T19:00:00+0100",`
		`"pwd-hash": "AQIDBAUGBwg=",`
		`"salt": "Mq7wFw==",`
		`"hash-function": "sha-512"`
		`}]`
		`}`
			Pre authentication

MQTT Authorization ACLs (Hono-TODO):

- can publish under: [t,e,a,r]/tenant/#
- If not gateway, can publish under: [t,e,a,r]/tenant/device-id/#
- If gateway, can publish [t,e,a,r]/tenant/child-devices/#
- Can subscribe only to: c/tenant/device-id/#

Other MQTT Protocol Adapter Assumptions:

- clean-session MUST be true. Connect requests with clean-session=false will be rejected.
- retain flag MAY be true. Messages will be confirmed if necessary, and an “x-opt-retain” application property will be added.
- Messages published with qos=2 are not acknowledged, discarded and tracked in malformed-messages Hono metric.
- LWT will be ignored if provided. Hono will log it and track it in “some” Hono metric.

Service Integration - Authentication (P2C)

Authentication API - Used in the Platform-to-Cloud (P2C)

Decision: For the purpose of Hono + Kapua integration, support for the Hono Authorization API in the Kapua Provider will be addressed in a Phase 2.

Service Integration - Connectivity API

Hono-TODO: Add Connectivity API for life-cycle messages

Notifications on connection established are guaranteed.

Notifications on connection losses are best effort. (Hono-TODO: is there a way to guarantee those notifications as well?)

Berlin Namespace - Digital Twin Namespaces

Hono Device Registry

- Each device has an ID
- One device can be registered “via” another device

Kapua+Hono

- Kapua will registry top-level devices only. Kapua Assets will not be registered in the Hono Device Registry

References

MQTT

https://esf.eurotech.com/docs/mqtt-namespace
https://www.eclipse.org/hono/user-guide/mqtt-adapter/
http://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-mqtt-support
https://docs.aws.amazon.com/iot/latest/developerguide/protocols.html

Digital Twin

https://esf.eurotech.com/docs/kura-wires-mqtt-namespace
https://docs.aws.amazon.com/iot/latest/developerguide/device-shadow-mqtt.html
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins
<Eclipse Ditto>

Review - Device-to-Cloud (D2C) Messaging

Review - Hono / Kura Adapter Message Mapping

Kura Message Type	Kura Topic	Kura QoS	Hono Address	Hono QoS	Notes
Control Messages	$EDC/tenant/device/	QoS=0,1	event/tenant/device/	QoS=1	
	$EDC/tenant/device/	QoS=2	DISCONNECT		
Data Messages	tenant/device/	QoS=0	telemetry/tenant/device/	QoS=0	
	tenant/device/	QoS=1	event/tenant/device/	QoS=1	
	tenant/device/	QoS=2	DISCONNECT		

Review - Cloud-to-Device (C2D) Messaging

Review - Command Control / Request-Reply Message Conventions

Message Type	**Kura Topic**	**Kura QoS**	**Hono Address**	**Hono QoS**
Device Subscription	$EDC/tenant/device/#	QoS=1	control/tenant/device	QoS=1
Request Message	$EDC/tenant/device/[application/verb/resource] Payload with requestId	QoS=0	control/tenant/device Message Header with message-id	QoS=0
Response Message	$EDC/tenant/device/application/REPLY/requestId		control-reply/tenant/device Message Header with correlation-id	

Open Questions

- Identity of connected devices, gateways and assets. Common picture and glossary.
- How to propagate asset identities through device identities through Hono
- Hono Federation
- Multicast to Device Groups
- Assert Credentials API (case: authentication through GitHub, Google, etc)