

Monthly Architect's Meeting, October 18, 2021, 10am PDT

Attendance:



Some attendees may have joined after the call started when this record was captured.

High Priority

- ADR reviews
 - [Service List](#)
 - Per last month's meeting, architects were to review, in addition to the ADR itself:
 - [Registry ADR](#) (already approved)
 - Security ADR for creation of secret store tokens for delayed start services (<https://github.com/edgexfoundry/edgex-docs/pull/389>)
 - During this monthly meeting make a determination about what to do about service list.
 - Address per recommendations in the ADR that have been approved by several architects – potentially creating a more detailed design (addressing secure and not secure)
 - Do nothing (allowing what is in the approved registry ADR and what new elements have been provided in the security ADR to override this ADR)
 - Update the ADR and take a different approach
 - Decision:
 - **Reject ADR**
 - Security ADR (398) – addresses addition of new service that needs a token. We already had a way to know what existing services need the token. Bottom line – services will get their needed tokens.
 - Proxy Setup – proxy setup knows existing services via env var; use the admin API to add / delete services / “custom routes”
 - CLI – new CLI knows existing ports of existing services; but does not address adding of new services
 - UI – that does an API call into registry for “x-service” – requires Consul; calls on metadata for device services

- SMA – totally dependent on code
- Rules for current and future consideration
 - Consul running – you use Consul API to get list of running services; use callback or make another pull to get updated list
 - Services need to implement the full Registry ADR to know about changes in services
 - Consul not running – use a configuration file but we don't have an unified, global configuration file for this (or a way to seed it to all services)
 - In the future, it might be nice to have a service TOML list that is provided to all services or is something metadata is provided and gives to all services.
 - May also look into something like DNS-SD service discovery

Meeting end and did not get a chance to discuss these topics

- [metrics](#)
 - Per last month's architect meeting:
 - Jim reviewed go-metric capability (now written in the ADR)
 - Jim updated the ADR to address Ireland release (state of SMA, OMQ, config changes around message bus, etc.)
 - During this meeting, discuss path forward.
 - Use of a library or home grow for our needs
 - Size impact to services
 - Go vs C implementation
 - Whether the telemetry needs to be persisted?
 - Does the telemetry need to be queried at the edge (with an available API)?
 - Do counts need to be reset?
 - Global configuration is ruled out (address this in the consequences). However, a global configuration ADR is needed for EdgeX 3.0. **Jim to create the issue.**
 - This ADR to be re-reviewed at the next Monthly Architect's meeting.
 - For records sake, Jim also added an issue for someday creating an ADR for global configuration
- [UoM](#)
- Declarative Kong applicability
 - Allowing us to drop Postgres DB
 - But can you configure groups/users ACL
 - Only support JWT users
 - Lose some functionalities we need: it would make the API Gateway read-only, and users can't create new users and tokens dynamically.

Medium Priority

- Med - Is the Wiki the best place to document project (low level architecture or design) decisions (those outside of or smaller than ADRs). This was our initial take. Should we revisit? (Per Jakarta planning meeting)
 - Jim to research other projects and how they handle this
 - Completed
 - K8s: has a architecture SIG, meetings recorded and put in Google Docs
 - Often create PR for issues and debate in the PR
 - <https://github.com/kubernetes/community/blob/master/sig-architecture/README.md>
 - TensorFlow – uses process of RFC (request for Comment) that get submitted by PR to rfc directory in Github
https://www.tensorflow.org/community/contribute/rfc_process
 - React Native – repository and PR process (but doesn't appear to be very active) <https://github.com/react-native-community/discussions-and-proposals>
 - Most use Github: Visual Studio Code, Google Flutter, Apache Spark, Kubernetes, Pandas, TensorFlow, PyTorch, node.js, vue.js.
 - ADR template is well used but where people keep these is not consistent and at what level an ADR is required is not consistent
- 3rd party lib review process
 - Per planning meeting; relook policies for 3rd party libs and how to evaluate, especially for device services. Also, should we start to fork these “immature” libraries

Low Priority

- Per Core WG meeting of 4/22/21 and Jakarta Planning Meeting (6/21) - how much info should be put in errors and log messages.
 - There is a concern about putting too much information in from those familiar with commercial products
 - Temporary decision was to provide enough so that someone could debug the problem with the information provided and not to be concerned with exposing intellectual property since everything is open source.
 - Per planning meeting: need a survey to address but don't believe we have an issue
 - **Potentially is an issue in defining what is Error/Info/Warning**
- Revisit combine core services at least at all executables in one image
 - Release would be easier but image would be bigger with more complex compose files
 - Per Core WG of 2/18/21 - is it at least worth exploring the combination of Core Metadata and Core Command since the two have to share so much data?
 - Core command is just a proxy service today, but reasons for having a separate service include: additional security to protect actuation; issue multiple device commands with one request (make one request and fire it to all Modbus devices or all devices under the control of one service); provide the means to limit requests down to a device so as not to overwhelm it or wake it up). These needs could also be incorporated into a combined metadata service but there are advantages to separation of concerns.

- Time series database support and applicability
 - Ian Johnson has an example of app service to InfluxDB export (snap in the store)
- Where should tool/script for creating new device and application services be placed?
 - After the architect's meeting of Jan 26, 2021, it was decided that "templates" should be created in all SDKs to allow for the easy creation of new services (removing the old samples in the SDKs). The templates will be a means for users to copy and create a new service with some instruction on how to rename and replace TODOs with necessary code.
 - After the templates are in place, there is a decision to be made about where automation can be placed to use the templates to create new services (versus a manual copy). In the CLI, in a new tool, in a set of simple scripts?

Tabled for now

- V2 API - should we add security foundation added to that (per some of earlier V2 API designs via Dell and Bryon N)?
 - Adding token to authenticate a micro service call (is this in scope for Ireland)
 - May not be needed unless all services are distributed
 - We need to explore alternatives to provide secure / locked out service to service communications
 - **ADR being created and to be reviewed in the Security WG.**
- Digital twin (and LWM2M) applicability
 - **Being worked via liaison with DTC**