

# Architecture Monthly Meeting

Dec 4, 2019

Organizer/host: Jim White (IOTech)

Attendees: Jim, Iain, John L, Steve (IOTech), Michael (LF), Lisa, Bryon, James, Lenny, Mike (Intel), Akram, Michael, Tingyu, Trevor (Dell), Dave, Rodney (Beechwoods), Colin (Kong), Tony (Canonical), Andre (Redis Labs). Note others may have joined after the meeting opened that are not recorded.

This was an extra Architect's meeting to alleviate some backlog and address two critical issues sooner in the Geneva dev cycle.

Topics to be discussed

- Lenny's go-mod-registry design and associated issues related to config bootstrapping
  - <https://github.com/lenny-intel/edgex-go/blob/RefactorRegistry/adr/Registry%20Refactoring%20Design.md>
  - Decision: have Lenny clean up a few items in the document; specifically, around:
    - Add comment on logging where config is coming from
    - Change stem to base\_path
    - Add section on get config endpoint for system management
  - Otherwise architects generally approve what is in this document and return it to the core working group for final review, approval and adoption.
  - In addition to Lenny's doc, conversation was added around IOTech's config proposal:
    - 1) We drop the use of config-seed
    - 2) On initial bootstrap of the service, the following configuration is loaded
      - a. Hard coded configuration is made part of each service and is the default configuration
      - b. Command line arguments override hard coded config values
      - c. ENV variables override command line arguments and hard coded configuration
    - 3) Configuration collected through above means is loaded into the configuration mechanism of choice (Consul or config file) as specified by some command line arg on initial bootstrap. In the case of a file, a new configuration file is written.
    - 4) The configuration mechanism of choice (Consul or config file) is used thereafter to get all config on service restart (the service will have to check if Consul is populated or the config file is present based on the configuration mechanism of choice command line indicator).
    - 5) To change the configuration of a service after initial bootstrap, one must change the configuration value in the configuration mechanism of choice (Consul or file). Hard coded config, env vars and command line vars would all be ignored (save the one command line arg that dictates where the config comes from).
    - 6) Should a user wish to clean the configuration and set the service back to its initial bootstrap state and cause the service to redo the initial

bootstrapping, the configuration in Consul or the file would need to be removed and the service restarted. Optionally, it might be nice to have some sort of command line flag to wipe the config by again going through #2 above.

- While there is some support for getting rid of config-seed, there are issue around other elements of this proposal – specifically where config should be hardcoded into the services, not addressing HA types of concerns and more.
- A request was made for a more thorough design doc to be created and worked through the core working group. Jim to take lead on this.
- Proposed GitHub projects management process (Bryan N, Lisa, security WG)
  - Bryon present info around suggestions in:  
[https://docs.google.com/presentation/d/1PghmwA9HJ\\_dJFEUG1-UPfsGYma\\_QR5QcMAHogSxdgYU/edit](https://docs.google.com/presentation/d/1PghmwA9HJ_dJFEUG1-UPfsGYma_QR5QcMAHogSxdgYU/edit)
  - This is being experimented with in security WG already and being considered for wider adoption.
  - Feedback and several suggestions on refinements were made during the call.