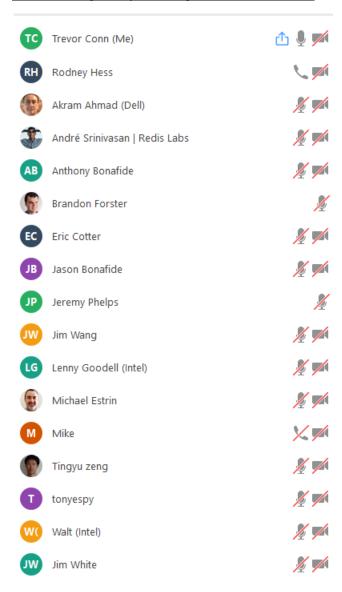
Core Working Group Meeting Notes (5-Mar-2020)



- 1.) Demo of rogue device "blacklist" handling via Kuiper (Jason Bonafide @ DELL)
 - a. Create ADR including links to code samples (holding) and diagram
 - b. Initial community approval to utilize this mechanism for blacklist capabilities in Device Discovery (Geneva)
- 2.) Config-seed removal dependency chain
 - Approve Jim's ADR
 - Go-mod-bootstrap ability to self seed config (Lenny)
 - This is a pending capability for DS and App SDKs to adopt go-mod-bootstrap
 - Integrate go-mod-bootstrap into all services
 - o More than just consuming the latest rev of the module, code changes required
 - Lenny has an example for App SDK

- Changes to core/support services to self seed
 - Lenny also has a example branch here for how to accomplish this
 - Edit all Dockerfiles to remove –profile cmd line
 - Remove res/docker directories
- Update docker-compose file
 - Includes environment variable overrides previously set in docker profile
 - Remove config-seed entry
 - o Remove config-seed entry from SNAP
- Remove config-seed
- ** Current rules engine will not work in current form after the above b/c it won't self-seed **

Discussion items:

- Is Kuiper the replacement?
 - Level of confidence? HIGH
- Backward incompatibility?
 - o Leave config-seed in place to just seed existing RulesEngine? -OR-
 - Include configuration in RulesEngine container, set process to always use local configuration and not Consul
 - What about configuration variance during deployment?
 - JIM WHITE Audit Rules Engine configuration on viability of always using local config in the Docker image.
- 3.) Error Handling Improvement Suggestions
 - a. https://github.com/edgexfoundry/edgex-go/issues/2403
 - b. Timing of incorporating suggestions? V1 or V2?
 - i. Trevor's preference is V2.
- 4.) Discuss whether configuration changes included in PR 2401 constitute backward incompatibility.
 - a. Time permitting this week
 - b. The question is whether or not a change of this kind breaks backward compatibility for Device Services / App Functions and, if so, does this PR set a bad precedent. I think there's agreement it's fine for the Core Services.
 - c. Trevor to refresh memory on actual code context of usage
 - i. Proposal from the community is to support both, not real enthusiastic about that
 - ii. Community also seems to feel this should be considered a backward breaking change