Core Working Group Agenda 4/5/2018 meeting

Attended by: Tony (Canonical), Rodney & Brad (Beechwoods), Keith & Andy (IoTech), Jim, Trevor & Tingyu (Dell), Janko (Mainflux), Jeremy (LF), Dan Bryant. There may have been in attendance later in the meeting after the list of attendees were assembled.

Discussion and action items as a result of meeting in RED.

EdgeX Awards: nominate someone today!   https://goo.gl/forms/Pk4GcJ7amSPgtyVJ2

Or @ https://wiki.edgexfoundry.org/display/FA/EdgeX+Awards

Old Business

DevOps issues/updates
- Automation of black box tests is priority 1
- Go & Arm CI (64/32 bit)
- Build 0.6.0 of EdgeX: suggest April 20th for HM event

Current Work Task Updates
- Work items are in Github Projects/issues.
- Drasko created Projects/tags/Kanban for California (and Delhi). See
    - https://github.com/edgexfoundry/edgex-go/projects
    - https://github.com/edgexfoundry/edgex-go/projects/2?add_cards_query=i

Documentation
- Game plan:
    - Github documentation eval available:
        - Docs going to be hosted under docs directory of Go mono repo in EdgeX Github the raw markup files (available Apr 9)
            - Andy to tap Jeremy when PR ready
        - Using IoTech Website to host for now - the finished documentation product for people to eval - with temporary link on EdgeX Wiki pointing to IoTech site (availablity to be determined)
            - Andy to coordinate with Jim on link from Wiki
            - Work with Brett on keeping track of Wiki page changes during eval period
    - Steve working on build process for documentation; will be integrated into Jenkins process/CI
        - Jeremy: where / how does Docker play a role?
        - Discussion to be started and worked off line.
    - Target early May cut over from Wiki to new documentation.

DS Requirements Discussion
- Open issues in DS requirements
    - Query All command results
    - logging/scheduling - *inprogress*
    - actuation commands RAML accuracy - *tbd* may be resolved
    - metadata updates (/callback) RAML incomplete - *tbd*

- o finalize appendix A on settings – does current settings/config work take care of this?
  - No updates this week; updates coming next week

License file issue

- How do we get appropriate license files to project artifacts (Docker containers and such)?
  - Copying workflow from work by Tony on Snaps. Jeremy to implement; will require a change to the Docker files.
  - Export services in Go have no attribution file. Jim to create an issue for tracking; Janko to handle.

Testing

- Andy leading sub-project under Security WG for security testing – meeting being setup for next week. Contact Andy or Brett for participation details.
- Additional blackbox tests – call for participation
- Performance testing – pushed to Delhi, but what do we need to start to consider (Andy was going to start considering)

Service Naming, Availability and Configuration Bootstrapping

- Quasi requirements/design doc available at: https://wiki.edgexfoundry.org/display/FA/Architecture+Issues+and+Decisions#ArchitectureIssuesandDecisions-ServiceIdentifiers.
- It addresses:
  - How do we name services?
  - How is service availability determined (for startup and use)?
  - How is configuration (to include service name) to services provided (consistently) and how do we handle the profiles?
  - How do we prepare (in small ways) for multi-instance services?
- Trevor and team are implementing (to include config-seed in go changes)
- Will need Application WG to take changes and apply to export services
- Everyone is asked to review and provide feedback for adoption next week.
- Tony to apply to DS requirements and SDK design/implementation
- Tony – this should address some open Issues and PRs in export and other services on how this should be handled (see Issues #105, 106, 107 and PR #119).
- Do we need to allow for per-property command line override (like they have in Java)? Jim to add to Delhi considerations.

New Business

Contribution Guidance Top 3 (or 4) Do's & Don'ts on PR – a recommendation for the Wiki and top of the contributions.md

1. Commit subject line messages should explain the problem that the commit is solving in 50 characters or less. "Fixing bug" or "adding comments" doesn't explain the problem the commit is solving!.
2. Provide a message body to the commit to provide more detail explanation. When you do, keep lines in the body to 72 characters or less.

You can add a commit subject line and multiple message bodies like this:

git commit –s –m "This is the subject; keep to 50 char" –m "This is the first line of the message body"$'\n'"This is the second line of the message body."$'\n'"Keep all message lines to 72 char."

3. Commit your changes in logical chunks. Make it easier for the community to review, comment on, and accept your contribution. For example, create a separate commit for a big fix and an enhancement.
4. Locally merge (or rebase) the upstream development branch into your topic branch. Do not merge master into your local PR branch (this is the lazy approach to dealing with committed code locally in a branch, and then realizeing that your PR isn't merge-able because HEAD has changed). See https://git-scm.com/book/en/v2/Git-Branching-Rebasing if you need more help.

- Tony going to provide revised wording to #4 above.
- Asking for communities input by next week.
- We'll append to top of Contributors.md and top of Contributors page on Wiki when approved (hopefully next week).

Other items

Tony – with new email system, the email lists are camel cased. This caused email filter problems for some. Jim to check with Jeremy/Brett @ LF if these will remain the same now.

Jim – ask that everyone start to consider areas of architectural concern/technical debt that we want to address in upcoming F2F and start to work on for the Delhi (or later releases). Start to cover this in upcoming meetings (at least the planning and pre-research we want to have in place before June meeting).