Security WG Meeting, 6/19/19

Attendees: Jim, Anthony, Brandon, Trevor (Dell), Jim Wang, Mike,Lenny, Bryon (Intel), Malini (VMWare), Rodney(Beechwoods), Ian, Tony (Canonical), Ike . Others may have joined after the meeting started and attendance was captured.

Agenda

Old Business

- Security Issue Process
    - Updates from Malini
    - Issues list: Web page/Wiki page/Docs location
    - What's next -> implementation
        - ✓ Email address
        - ✓ Web page setup
        - Initial issues list
            - Where should it exist? Github?
            - It's a working live document (verses snapshot)
            - We need to have separate list of each release (static)
            - Then a place where people can monitor those discoveries post release (dynamic)
            - Could Wiki work?
            - Proposal: start with Wiki (in table). Allow it to be searchable.
                - Summary of issue -> with link to (optional) Github issue (with more details) or whereever the CVE issue lives
                - Put link on Website to Wiki
        - Formation of the SIR
            - Looking for people to serve on this team of 3-4
            - Doug Gardner has volunteered
    - On hold: Exploration of improvements to this process (moved to next WG chair agenda)
        - Malini has reviewed and incorporated best of breed solutions into our docs
        - Offer from Kate Stewart from LF and Zephyr project
        - https://cve.mitre.org/cve/request_id.html#cna_participants
        - Aware of gaps/issues, please provide Malini details.
        - Issues to be addressed in next version
            - Should our policy address releases and when we release when bug fix hits a certain level of security issue.
            - How does Kubernetes and other projects do it?
- Service Implementation Updates from Tingyu
    - Secret Store Service work – complete (just need PR merged)
        - Updated to Go 1.12
    - API-gateway structure to be consistent with EdgeX core projects – completed (just need PR merged)
        - Updated to Go 1.12
    - Integrate client with Secret Store Service (go-mod-secrets) - completed

        - Work with DevOps on CI/CD
    - Still to be done
        - move security API Gateway / secret-store into edgex-go (the "mono" repo)
        - Update docker compose files (making security version the default)
        - Move MongoDB init to Go – started (acceleration of effort)
        - Add testing cases and docker compose file in black box testing
    - Schedule – at Tingyu/Malini discretion
        - Considering earlier freeze date for Fuji
        - Work complete for dot release but work group feeling it should just be part of Fuji
        - Does work constitute non-backward compatible change in EdgeX (requiring v2.0)?
        - How do we upgrade – how do we provide instructions to users. Possible scripts - update ports, users, etc.
- Other Fuji work
    - Generation of PKI – Bryon/Jim Wang have started
        - Distribution of per service Vault secrets
    - HW secure storage abstraction layer (design only) – Bryon/Jim Wang & Malini
        - How to protect the Vault Master Key
        - Suggested to have design meeting – possibly in Palo Alto
    - Ensuring the services running are those expected (and authorized) - Malini
        - Design/approach
        - Need some crowd-source help
        - Need solution that covers 3rd party creation
        - Need to capture use cases
        - Is this just about signing packages or about runtime as well? Is it just solved by signing Docker images (for example)?
        - Is something internal to EdgeX a core competency? Is that something we want to do?
        - Need to cover install and launch
        - Maybe it is tied to something that distributes tokens to the right things
        - First – defining Scope/ what use cases are we attempting to address
    - Renew/refresh threat assessment - Tingyu
    - Need document defining what security is/does and can/will do – Jim White/ Tingyu
    - Application Services will need access to Vault (through client and Secret Store Service) for tokens/certs/secrets for HTTPS/MQTTS connectivity, cloud access, etc.

New Business

- Self-assessment & security related issues; bootstrapping with someone else's system (Certification WG) – Rodney
    - We need to do self-assessment of their work in a secure environment.
    - How would that work? – with regard to Device Services (targeting first for self-assessment)
        - Ex: Developer has a set of stubs that mimics services
        - Ex2: we have some staged environment that allows them to test in

- Is it not a variation of black box testing?
    - What do device service black box testing?
    - Then what do device service black box tests look like for the secure black box testing?
    - For self-assessment, we’ll need documentation around how to pull down black box tests (secure and unsecure) and how to run against services (with and without 3rd party service in place).