Core Working Group

June 04, 2015

Reminder:
This call is being recorded

Antitrust Compliance Notice

- AllSeen Alliance meetings involve participation by industry competitors, and it is the intention of AllSeen Alliance to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of and not participate in any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

- Examples of types of actions that are prohibited at AllSeen Alliance meetings and in connection with AllSeen Alliance activities are described in the AllSeen Alliance Antitrust Policy. If you have questions about these matters, please contact your company counsel, or if you are a member of AllSeen Alliance, feel free to contact Lee Gesmer or Andrew Updegrove, of the firm of Gesmer Updegrove LLP, which provides legal counsel to AllSeen Alliance.

Agenda

1. 15.04a Decision

2. Proposal to remove deprecated version of SessionListener::SessionLost in 15.04a

3. 15.08 Features

4. Task prioritization in JIRA

5. SRP deprecation

6. Supported bindings

7. Samples Proposal

8. Review Action Items

15.04a Decision

- Proposal by MSFT to move decision to ship out by one week

- Core WG decision will be made on June 11

Proposal to remove deprecated version of SessionListener::SessionLost in 15.04a

- Proposal from MSFT
  - The SessionLost(SessionId sessionId) callback has been documented as deprecated for a long time. It has been superseded by the newer SessionLost(SessionId sessionId, SessionLostReason reason).
  - In an effort to support both versions of SessionLost, the current implementation of BusAttachment silently blocks apps from destroying their SessionListener object during SessionLost. BusAttachment ends up using already freed memory if the app choses to destroy its SessionListener from the SessionLost callback – see ASACORE-2048 for the relevant source code.
  - But SessionLost is the proper place for destroying session-specific data. Therefore I propose removing the old SessionLost(SessionId sessionId) from SessionListener.h, and continuing to support just the newer version of SessionLost, in 15.04a. Apps will have to use the newer callback version when/if they get recompiled.

- MSFT states
  - We have concluded this to be a must fix for 15.04a. This is low risk change but given the time need for honoring the deprecation process etc., the earliest this fix will be merged is tomorrow (?) which means we will likely need to move the patch release date by a day to give us time to integrate this with Windows 10 build and run some basic sanity validation. Just wanted to send a heads up email for the need for potential change in the patch release date and see if there are any objections from your side.

- Core WG Decision
  - Deprecate, update release notes, update TSC
  - Fix should include updating samples

15.08 Features

- Need to firm on feature list from all contributors
  - https://jira.allseenalliance.org/issues/?filter=11008
  - Current committed features
    - Security 2.0 (MSFT, QEO, QCE)
      - https://jira.allseenalliance.org/browse/ASACORE-1393
    - Commercialize UDP Transport for TC <-> RN connections (QCE)
      - https://jira.allseenalliance.org/browse/ASACORE-1686

- Expectation is to close on the feature list by June 4
  - Communicate to the TSC after feature list is firmed
  - Open items will move to 16.04
    - ASACORE-1430 (MSFT)
    - ASACORE-1166 (QCE)

Task prioritization in JIRA

- https://wiki.allseenalliance.org/core/overview/jira_process#draftsetting_severity_for_tasks

- Proposal to use severity for tracking task priority for a release
  - Minor – nice to have
  - Major – desirable, but best effort
  - Critical – highly desirable; wont block release if incomplete
  - Blocking – required; will block release if not completed

- This has the following benefits
  - Easier tracking of tasks required / desired for a release
  - Keeps severity and priority separate
    - Severity relates to the code base / release
    - Priority relates to workload management

SRP deprecation request

- From Greg Zaverucha (MSFT) posted to Core WG mail list on 5/8/15
  - SRP is an authentication mechanism in AllJoyn that allows two endpoints to establish a shared secret knowing only a password. Unlike ECHE_PSK, the pre-shared value in SRP may have low entropy (like a 4-digit PIN printed on the box of a Thing). This is a useful feature, but SRP is not the best way to realize it, for the following reasons
    - SRP is not supported on the thin client. It's also not a good fit for low-power devices, as it's computationally more expensive than elliptic-curve (EC) based primitives.
    - SRP uses a separate crypto stack from the ECDSA and ECDH. To implement SRP requires large integer arithmetic (bignum) and SHA-1. SRP is only auth mechanism using these. (the others share core the EC code, and use SHA-256).
    - The current SRP implementation needs work, it lacks protections against side channel attacks. The considerable investment improving it is better spent elsewhere.
  - In place of SRP I will propose an alternative protocol to get the same functionality that is a good fit for the thin client, i.e., use the same underlying EC code and hash function. I am currently reviewing the password-authenticated key exchange protocols specified in IEEE 1363.2, one of these protocols will form the basis of my proposal. The design and implementation of the new mechanism will target the 15.08 release.

- Tentative Core WG agreement, no opposition after 1 week of requesting feedback

- Note: This will need to be presented and voted by the TSC on June 11

- Next steps:
  - MSFT to understand the ecosystem and impact before making proposal (reach out to LIFX)
  - Chris (QCE) to move this to the action item list

Supported bindings

- Discuss policy for supported bindings

- Existing bindings in Core
  - C++
  - Obj-C
  - Java
  - C
  - Javascript NPAPI

Samples Proposal

- Current samples are inconsistent and not using best practices

- Proposal
  - Leverage Gavin's toaster sample
  - Remove the old samples

Action Items (1/2)

- Chris (QCE) to schedule
  - 15.08 backlog part 2 – 1pm PT 5/28
  - 15.04 post mortem – 1pm PT 6/8

- David (QCE) to lead team crafting testing proposals
  - Board approved budget
  - Charters crafted for RFP committee and test policy committee and sent to mail list for review

- Marcello (QCE) to send IOS language binding proposal to WG mail list
  - Agreement is to stop supporting the language binding
  - Todd to see if Vlad wishes to take ownership of this binding
  - Next step: Inform TSC

- Changing Windows to TCP/9955
  - Dan (MSFT) will send mail to Core WG, discuss, and TSC mail list about this change going into 15.04a
    - Dan (MSFT) to add to compatibility list for 15.08
  - Marcello (QCE) will add this to the release notes.

Action Items (2/2)

- Proposal to outline the process for changing APIs
  - Gavin (MSFT) will craft proposal for review by Core WG and then presented to TSC
  - Will review document by 6/11

- Dan (MSFT) to submit proposal to Core WG mail list about possible deprecation of ajn::BusAttachment::Connect(connectspec) overload
  - Most likely for future release
  - Dan (MSFT) to create JIRA ticket

- Dino (MSFT) to understand the ecosystem (LIFX) and impact before making proposal aboutSRP deprecation request

- Chris (QCE) to create JIRA task to define a manifest and add a check to identify delta
  - https://jira.allseenalliance.org/browse/ASACORE-2038

Thank You

Follow Us On [facebook] [twitter] [rss] [linkedin] [google+] [youtube]

- For more information on AllSeen Alliance, visit us at: allseenalliance.org & allseenalliance.org/news/blogs

Language bindings discussion 5/7/15

- Need a formal policy to support language bindings
  - Came up at last TSC F2F

- SCL binding proposal
  - Required
    - C++ and C
  - Optional
    - Java, NPAPI, ObjC?

- TCL binding proposal
  - No proposal
  - Should we consider JavaScript?

- Need separate policy for platforms
  - May need to consider binding platform and language binding

- Next steps
  - Action: Marcello (QCE) to send proposal to the mail list

Notes from 14.12 Post Mortem

14.12 Post Mortem Improvement Items (1/3)

- Aligning date & the end game (lockdown) schedule of AllJoyn releases with release schedule of the contributing member companies if it happens to be in close proximity of AllJoyn release
  - **Action**: Arvind to send proposal to Core WG mail list

- Consistent and enforced definition/bar for code freeze
  - Need crisp definitions for "incremental bug bars" (normal, tell, ask)
  - Need approval granularity (approval on merge)
  - Need process for how to deal with large last-minute changes
    - Suggestion – only high priority issues "ask" are added one week before release
  - **Action**:
    - Marcello to send current milestone definition to Core WG mail list
    - Gavin to send proposal to Core WG mail list based on Marcello's email

- Need processes for breaking changes
  - Regarding protocol, API syntax, behavior
  - Mitigation: Proposed changes should be advertised
  - **Action**: Chris to add the process to this to the existing process draft

14.12 Post Mortem Improvement Items (2/3)

- Need agenda and slides 48 hours ahead of core WG meetings
  - Best effort to send slides by COB Friday

- Need notes from core WG meetings sent more consistently
  - Note: Linux foundation unable to assist
  - **Action**: Gavin to see if someone from Microsoft can assist

- For TSC: PR coordination for releases
  - More an issue for marketing committee
  - **Action**: Chris to discuss this with Philip

- Engage system test during feature testing
  - More members conducting system test is preferred

- Testing needs to be better distributed across members

- E2E testing is needed
  - **Action**: David and Arvind will make a proposal

- More frequently merge feature branches so that deltas can be kept to a reasonable minimum
  - **Action**: Chris to add process to the Wiki process draft

14.12 Post Mortem Improvement Items (3/3)

- May need more frequent but shorter Core WG meetings
  - WG meeting immediately following triage meeting for 30 minutes
  - **Action**: Chris to change Core WG status meetings to 30 minutes after the Thursday triage

- Increase frequency of triage meetings earlier in the process
  - **Action**: Chris to set up biweekly triage 2 weeks prior to branch date
    - For 15.04 it will begin week of March 9

- Define JIRA severity vs. priority process
  - **Action**: David & Arvind to set up discussions to craft proposal

- To be discussed at next meeting
  - JIRA label to identify contributing organization taking ownership of the item
  - Need process for managing the platform matrix
  - Revisit code style guidelines, rules, and enforcement