ALLSEEN ALLIANCE

Core Working Group

May 26, 2016

The contents of this document, and the interfaces described, and all the information herein, are the result of collaborative discussions by the Core Working Group. This summary documents the final consensus of the team.

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

- AJATS discussion
  - ACTION: Evaluate the existing tests, and look at making them more robust, and make sure they test the right stuff.
  - ACTION: Josh to provide update on AJATS set up – in progress
  - ACTION: Marcello to look at slides Kris provided.
- Additional reviewers
  - Proposal is Krystian
- Krystian: ASACORE-2940 has been merged to master : should it be merged to some branch?
  - Agreement: close Jira, update rel notes, and back out the change from Master
- Pawel/Dan: Proposal to support the wildcard * anywhere in interface name, when using XML-based Security 2.0 APIs (ASACORE-2847)
- Pawel/Dan: Proposal to support interface member type = ANY, when using XML-based Security 2.0 APIs (ASACORE-2848)
- **Go Over 16.10 planning**

Near-term Release Planning

16.04 iOS Support and 16.10 Plan

16.04a iOS Plan (Affinegy)

Features/Tasks

- About infrastructure (ASACORE-2906)
- Unified Introspection XML
- Replace deprecated functionality
- Update documentation
- ~~AutoPinger?~~

Plan

Total Estimate:	8 +/- 2 weeks
Test Cycle Starts:	July 15
Code Freeze:	July 22
Project Complete:	July 29

- Carrie/Kris: More test infrastructure needed for 16.04?

16.10 Plan (Affinegy)

- iOS and Android
  - Security 2.0 bindings
  - Base Services Requests
    - ASACORE-2919, 2905
  - HAE Requests
    - ASACORE-1811, 2924, 2926
  - Other Requests
    - ASACORE-2772, 2748, 2744
  - Sec 2.0 C&C tests
- IPv6
  - Basic support for link-local discovery (MDNS) and reachability (TCP/UDP)
  - Stretch Goals: Support for site-local discovery and site-local + global reachability
- AllJoyn.js
  - Feature parity with Thin Core, including Security 2.0
- String cleanup?

AllJoyn.js Plan

- Needed for Gateway Agent 16.04 (part of Base Implementation)
- 16.04 Estimates (June 24)
  - Bindings: 2 weeks
  - Unit Test: 1 week
  - Documentation/Samples/OpenWrt feed: 1 week
- 16.10
  - Additional unit tests for Security 2.0
  - Support for additional features and changes to APIs

Discussion

- Is AutoPinger needed for Java and/or iOS?
  - Is it needed for 16.04?
- Does anyone have information regarding code generation in iOS?
- To what degree are the various Sec2 components (listed below) needed for mobile apps? What about for Thin Core?
  - PermissionPolicy::Acl, CertificateX509, AuthListener::Credentials, Crypto_ECC, DefaultECDHEAuthListener, DefaultPolicyMarshaller, ECCPrivateKey, ECCPublicKey, ECCSecret, ECCSignature, IdentityCertificate, KeyInfo, KeyInfoECC, KeyInfoNISTP256, PermissionPolicy::Marshaller, PermissionPolicy::Rule::Member, MembershipCertificate, PermissionPolicy::Peer, PermissionConfigurationListener, PermissionConfigurator, PermissionPolicy, PermissionPlicyUtil, PermissionPolicy::Rule, SecurityApplicationProxy, SigInfo, SigInfoECC, CertificateX509::ValidPeriod
- What mobile app and thin core samples will be needed for Sec2?
  - Replicate C++ app samples, Security interaction, claim a device, set a set of policies
  - Any other real-world samples we can think of

Documentation

- Core documentation needs major updates
  - Examples:
    - Everything is based on Ubuntu 12.04, but testing was done on 14.04
    - Many tutorials don’t work
  - Work has begun out of necessity (various members of team trying to check other areas)
- Do we want to do a full pass on documentation?
  - QA task – go through existing docs and file bugs (and possibly fix)

Action Items

Action Items

- Add a way in JIRA to track compatibility issues and proposals
  - Proposal is the use a label: “compatibility impact” in the Jira ticket
  - Committers should require for these sorts of change that the release notes be updated
  - **Marcello** to document this process in the Core Process wiki
- Define process to require regression/unit tests for bugs
  - **Marcello** to add description of Committers requiring unit test for bug fixes to Core Wiki
- Define process for handling “Technical Debt”
  - Example: Took a shortcut to make a release and not loosing track of this to fix the shortcut
  - Jira items to track when things are missing – e.g. feature added, but missing from Java binding
  - If shortcut is taken: could create a Jira task to track the problem

Thank You

Follow Us On

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