ALLSEEN ALLIANCE

Core Working Group

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.
  - ACTION: Josh (and possibly Marcello) to move forward with information from Lance
- Gregz: ASACORE-2880 Discuss possibility of CI support for core/test
  - ACTION: Arvind to follow up with Greg on the list of test programs that are useful from (
  - Kris/Carrie sent)
  - ACTION: Ry to file final bug on breaks in AJTCSCTest – done
- WayV: create list of issues related to UDP transport

Agenda II

- Kevin/Dan: ASACORE-1454
  - This contemplates offline distribution of policy updates and membership certificates. Currently, all Security 2.0-related operations have to be done over the network via method calls. One consequence of this is that there is a potential race condition when it comes to claiming, between when a new claimable app emits its State notification to when a security agent claims it, particularly if it can be claimed without authentication via ECDHE_NULL. But more interestingly, we may want an out-of-band mechanism to do management.
  - To that end, I'm proposing adding a set of APIs to the PermissionConfigurator class (which is currently used by apps to set their claimable state and set a manifest template, amongst other operations) that mirror the methods currently used by Security 2.0 management: the contents of the ClaimableApplication and ManagedApplication interfaces. This means Claim, UpdateIdentity, InstallPolicy, InstallManifests, Reset, and so on. This will then allow an app store to provision policy and credentials before the first time the app ever connects to the bus, and would let other apps/devices be provisioned by an out-of-band means, if such were desirable in their scenarios, or for testing purposes.
  - The proposed API will be a mirror of the current Security 2.0 API exposed over the network, projected into the C and C++ bindings.
- Agreement to move forward.

Action Items

Action Items

- Proposal to outline the process for changing APIs
  - Has been handed over to Brian Rockwell
    - will craft proposal for review by Core WG and then presented to TSC
  - Expectation that this will be reviewed with WG in January.
- Arvind (MSFT) to create Wiki process page linked off of Core WG wiki
  - Remove references to java from mandatory binding list for 16.04 timeframe
  - Hope to have something by the end of Jan.
- Ry (LF): Need to have Alliance running Windows 10 when RTM'd
  - Running the unit tests on the Windows 10 cannot seem to pass
    - Only one set of tests are failing, MSFT has worked with Ry.
    - Issue appears to be the combinations of Windows 10 and the version of VM ware
- MSFT: 16.04 work items, should have a sense by the end of Jan, or early Feb
  - Sec2.0 scenarios, working on C binding for Core Sec2.0

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