



**ALLSEEN
ALLIANCE**

Core Working Group

15.04 Post Mortem

6-22-15

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.



Reminder:
**This call is being
recorded**

What Went Well - MSFT

- Release followed schedule, 15.04 did not have last-minute breaking/unexpected changes
- 15.04a well scoped, not any "unexpected" or "un-needed" changes
- Process improvements from 14.12 were observed and put into practice
- Feature removal process was smooth
 - RSA, pin
- Increased Alliance testing coverage/scope
 - VS2012
 - Coverage
- Code reviews - turnaround time short
- Timing between release of 15.04, lockdown of Windows, //build event and announcements (although timing of 15.04a required changes beyond patch release, Ex: removal of deprecated method SessionListener legacy method)
- Collaboration and responsiveness MS <--> QCE

What Went Well - QCE

- Sticking to schedule / Planning seemed to be pretty effective
 - Security 2.0 was identified as a major risk early in the release cycle and pushed to the next release
- Feature branch approach worked
- Collaboration
 - Great participation from all contributors during meetings and mail list discussions
 - Broad scope changes went in smoothly
 - Good coordination and assistance with test
 - Contributors willing to assume test responsibility for introduced changes
 - Major new features from multiple contributors
 - More code review from multiple contributors
 - Lot of disparate but related changes that came together well
 - Crypto changes
- Router ranking feature was implemented early in the release cycle
 - Was well-tested well before the release.

What Went Well - QEO

- Security 2.0
 - Fruitful discussions for defining HLD concepts
 - Good collaboration and active involvement from all parties
 - MSFT security experience for defining HLD
 - Correct decision to postpone to 15.08 release
 - Code review process/Gerrit way of working
- Observer functionality in Core
 - Good interaction with core team
 - Found and tackled many relevant issues in core
- IRB
 - Very important for wide adoption and interoperability for AllSeen services of the future

What Didn't Go Well - MSFT

- Test coverage inadequate in some areas
 - Ex: Thread pool timing issues not found by tests (but unit tests that were added found these issues, would have been helpful to have had earlier)
- Documentation needs improvement
 - Developer guidance is not adequate or up to date
 - Check all documentation (readme.txt(s), ...) and review/update
 - ASADOCS bugs need more attention, quicker turnaround time, needs owner
- C samples are not up to date (need quality samples)
- Alliance unit tests don't run reliably against a Daemon router

What Didn't Go Well - QCE

- Integration branch was started too late.
 - Positive: There weren't many issues/conflicts between feature branches.
- Platform matrix communication
 - Last minute VS2012 confusion
- Shipping with known issues for 15.04
- AJTCL certificate bug
- Late breaking crypto changes
- Commit velocity was still high close to the release date
- Dependencies from other projects not managed well
 - Larger issue than Core
 - e.g. Base services build breaking
- Tracking compatibility information
 - Specifically updating the release plan page Compatibility section.

What Didn't Go Well - QEO

- Existing functionality being dropped due to lack of resources.
 - iOS, OSX
 - Java/Android not fully lining up with core functionality
 - Potential relevant issues postponed to later release
 - No clear ownership
- Unclear release cycle
 - What platforms, OS's are supported
 - Fragmented documentation
 - SDK layout
 - Samples not included in all relevant SDK's
- Security 2.0
 - Lots of time spend to define feature set and concepts, very confusing at the beginning
 - Initial development started without clear specifications
 - Not easy to integrate code in constant state of flux

Suggested Process Changes - MSFT

- C Bindings
 - Need clear ownership and responsibilities documented
 - Need better alignment on binding strategy all-up (AllSeen)
- Need to have Alliance running Windows 10 when RTM'd
- SCONS needs to support VS 2015
- Add ASADOCS for Core to the triage
- Check readme files as part of the release procedure
 - Core WG needs to decide who will own the specific readme files

Suggested Process Changes - QCE

- Set a project milestone to begin integration branch merges
- Track feature branches and add more spacing between feature branch merges
- Agree on platforms early
 - Already happening for 15.08
- Agree to feature sets and hold to agreed milestone dates
- Need more eyeballs on security for implementation and test
- Better implementation of unit tests
 - For things like the AJTCL certificate bug, make sure a unit tests are implemented
 - Consider adding process to require regression/unit tests for bugs
- Ensure people update release plan page when they propose breaking changes that are accepted.
 - Have begun with this for 15.08
- Add a way in JIRA to track compatibility issues and proposals
- Need to collaborate on stress/system testing
 - Add a track on system testing at F2F in October

Suggested Process Changes - QEO

- More robust automated testing
 - Investigation of occasional failures
 - Better feature/code coverage
 - Integration test scenario's covering multiple features on multiple platforms
- Actions already ongoing to improve release process
 - Sample cleanup
 - Better planning and tracking
 - Platform/SDK definition
 - Formation of Test Policy Sub Committee
- Process for handling “Technical Debt”



Thank You

Follow Us On      

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