ALLSEEN ALLIANCE

Core Working Group: Security 2.0

Aug 10, 2015

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

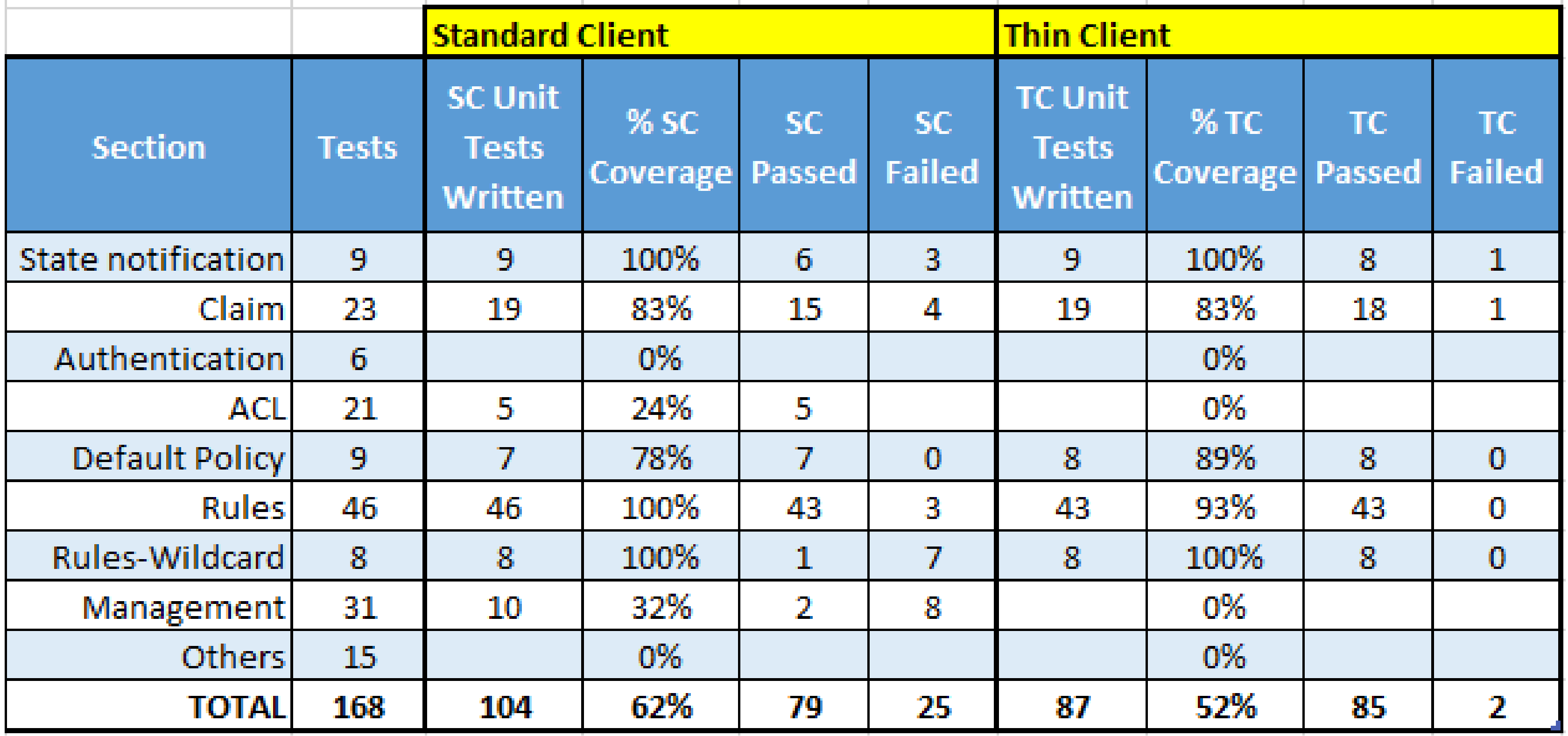
1. Schedule
2. Bindings
3. Triage
4. HLD updates
5. Discussion
6. Action items

Schedule

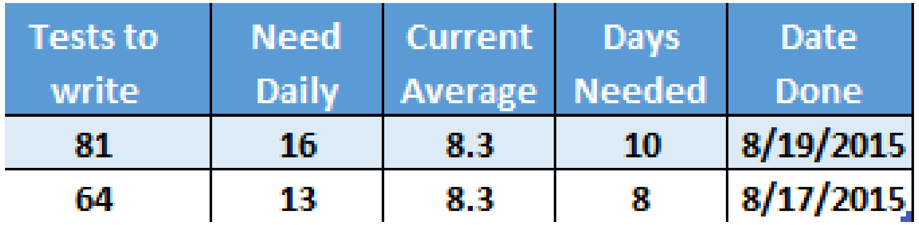
Schedule

- Dashboard: https://jira.allseenalliance.org/secure/Dashboard.jspa?selectPageId=10903
- Milestones
  - Merged with master
    - All changes should go to the master branch
  - 15.09 release – September 30
- Interop testing
  - QCE will conduct the documented backwards compatibility tests in the E2E section of the test spreadsheet.
  - MSFT will communicate the E2E testing they will conduct for Security 2.0

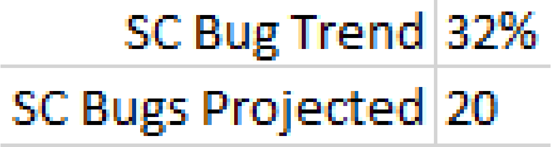
Test Case Coverage



		Standard Client				Thin Client			
Section	Tests	SC Unit Tests Written	% SC Coverage	SC Passed	SC Failed	TC Unit Tests Written	% TC Coverage	TC Passed	TC Failed
State notification	9	9	100%	6	3	9	100%	8	1
Claim	23	19	83%	15	4	19	83%	18	1
Authentication	6		0%				0%		
ACL	21	5	24%	5			0%		
Default Policy	9	7	78%	7	0	8	89%	8	0
Rules	46	46	100%	43	3	43	93%	43	0
Rules-Wildcard	8	8	100%	1	7	8	100%	8	0
Management	31	10	32%	2	8		0%		
Others	15		0%				0%		
TOTAL	**168**	**104**	**62%**	**79**	**25**	**87**	**52%**	**85**	**2**



Tests to write	Need Daily	Current Average	Days Needed	Date Done
81	**16**	**8.3**	**10**	**8/19/2015**
64	**13**	**8.3**	**8**	**8/17/2015**



SC Bug Trend	32%
SC Bugs Projected	20

Bindings

Bindings

- Core implementation
  - SC: C++
    - Gavin (MSFT) to verify internally if they will handle the C binding for Security 2.0
      - Will have more details by next week
  - TC: C

Triage

Triage

- Review unassigned tickets
- https://jira.allseenalliance.org/issues/?filter=11142

HLD updates

HLD Updates

- MSFT
  - Dave Thaler to update HLD for the membership privacy leakage change
    - ASACORE-2067 SC: information disclosure vulnerability with security 2.0 membership certs
- QCE
  - https://git.allseenalliance.org/gerrit/#/c/4612/ “Simplified ACL rule matching”
    - Waiting on +1 from MSFT
    - ASACORE-2194: Membership certificate definition is not correct
  - Identity delegation & certificate chain validation & EKU validation
    - Updates posted
    - Need a +1 from QEO
      - Joris had +1 this before Kane's comments and my changes to update them
  - ASACORE-2192: Certificate authority installed during claim is not manageable.
    - Status: Underway – Changes to 22 diagrams complete
  - GetAllProperties permission requirements
    - Short term: The consumer to only allow GetAllProperties if it has a matching PUBLISH ACL for the producer that includes * for interface member.
    - If it only has PUBLISH ACLs with interface members specified then that implies there may be members which the publisher is not trusted to publish therefore GetAllProperties should fail.

Discussion

Discussion

- QEO Discussion:
  - Manifest update scenario
    - Some possible improvements to the AllSeen framework that could improve the situation:
    - 1) Add a callback to the application that is called whenever the core receives an UpdateIdentity, that could be used by the application to see if the manifest has been updated and to reset the state the CLAIMED.
    - 2) Add a method to the ManagedApplication interface that could be used by the security manager to indicate that it has received the new manifest update and the application has been updated accordingly.
    - 3) Always comparing the ManifestTemplate with the template that is known to the security manager upon discovery of a remote application (and ignoring the NEED_UPDATE state altogether). This is the most robust, but introduces quite some overhead in network traffic that is in most cases unneeded.
  - Next steps
    - QEO to submit a feature request ticket

Action Items

Action Items

- None noted

Thank You

Follow Us On

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