Toward Hybrid Requirements-Based and Architecture-Based Testing of Security Requirements

Jose Romero-Mariona
Hadar Ziv
Debra J. Richardson
Donald Bren School of Information and Computer Sciences
University of California, Irvine
Irvine, CA USA
{jromerom, ziv, djr} @ics.uci.edu
Outline

- Motivation
- Our Positions
- Security Requirements Using the Common Criteria
- CCARCH: Our Approach
- Hybrid Testing
- Conclusions and Future Work
- Selected References
Motivation

- Security is increasingly important [1]
- Security is both a system property as well as a nonfunctional requirement
  - This makes it interesting
- Security should be addressed throughout the software lifecycle
  - Common Criteria lacks support beyond requirements

Our Positions

- Requirements-based testing of security concerns is possible
  - Security requirements are developed using the Common Criteria [2]

- Architecture-based testing of security concerns is possible
  - CCARCH approach is used to map security requirements into architectural components and connectors

- A hybrid approach is possible
  - Common Criteria for requirements-based testing
  - CCARCH for architecture-based testing

Security Requirements Using the Common Criteria

<table>
<thead>
<tr>
<th>Security Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
</tr>
<tr>
<td>Req1. Enforced Proof of Origin</td>
</tr>
<tr>
<td>Req2. Enforced Proof of Receipt</td>
</tr>
<tr>
<td>Cryptographic Support</td>
</tr>
<tr>
<td>Req3. Cryptographic Key Generation</td>
</tr>
</tbody>
</table>

Threats to Requirements Mapping

- Thre at1: Req1, Req2, Req6
- Thre at2: Req1, Req2, Req4
- Thre at3: Req1, Req2, Req3, Req7
- Thre at4: Req1, Req5, Req7
- Thre at5: Req5, Req6, Req7

CCARCH Approach: Security Requirements into Architectures

One to One Mapping

On

Many to Many Mapping
Hybrid Testing

QuickTime™ and a TIFF (Uncompressed) decompressor are needed to see this picture.

QuickTime™ and a TIFF (Uncompressed) decompressor are needed to see this picture.

QuickTime™ and a TIFF (Uncompressed) decompressor are needed to see this picture.
Architecture-Based Testing

The mapping from security requirements to architectural components can be verified using CCARCH v1.1.

- XML file describes each architectural component and its connections.
- During mapping verification:
  - If TRUE, it lets the user graph the initial architecture that is in the file.
  - If FALSE, the system alerts the user that there is a problem and points to the problem.

- The mapping of each architectural component is verified against its requirement.

Component is verified against its requirement.
Requirements-Based Testing

- Common Criteria-developed security requirements
- Systematically developed
- Confidence
- Include a variety of artifacts
- Allow for relationships to be easily traced
- Cohesive
- Provide support for testing
- Future work will be able to carry them further in the software lifecycle
- Traceability

QuickTime™ and a TIFF (Uncompressed) decompressor are needed to see this picture.
Requirements-Based and Architecture-Based Testing

- It should be possible, when Common Criteria and CCARCH are used, to
  - Test that components do not violate any security requirement
  - Test that connections do not violate any security requirement
  - Test that the architecture as a whole fulfills the security requirements as a whole

- Benefits to this testing approach,
  - Validation of requirements will help validate the architecture
  - Validation of architecture will help validate the requirements
  - Testing is enhanced using the information about the components’ connections
Conclusions and Future Work

Conclusions
- Security requirements expressed in Common Criteria can be used beyond the requirements phase.
- Our approach allows you to verify derived software artifacts against the initial security requirements.

Future Work
- Apply our approach to a bigger system
- Take Common Criteria security requirements past architecture
References

