“OMG: Not your Father’s CORBA Organization Any Longer”

The OMG System Assurance Task Force’s SwA Ecosystem

Dr. Ben Calloni, P.E. CISSP, CEH, OCRES Professional
Lockheed Martin Fellow, Software Security
Lockheed Martin Aeronautics Company, FTW
Lockheed Martin Representative to OMG
   OMG Board of Directors
   Co-chair OMG System Assurance Task Force

Djenana Campara,
CEO KDM Analytics
   OMG Board of Directors
   Co-chair OMG System Assurance Task Force
Defining Assurance

• Assurance provides the members of society a basis for believing certain assertions
• Assurance Processes provide the foundation for a belief system
• Assurance is the measure of confidence that the security features, practices, procedures, and architecture of an information system accurately mediates and enforces the security policy. - CNSS 4009 IA Glossary

• Dependability defined as the superset of availability, integrity, reliability, safety, and security -

People place “trust” in a system when dependability is demonstrably acceptable!
Delivering System Assurance (SysA)

- Assurance is 3 step process
  1. Specify Assurance Case
  2. Obtain Evidence for Assurance Case
  3. Use Assurance Case to calculate and mitigate risk

- Historically, Security Assurance is Informal, Subjective & Manual

No Perfect Safety/Security
Every System will have Residual Risk
The Goal

• Key Challenges
  – **Objective and cost-effective** assurance process
  – **Reduce ambiguity** associated with system weakness space
  – **Systematic coverage** of the weakness space
  – **Effective and systematic measurement** of the risk

**Modeling Analysis** to achieve high confidence in system trustworthiness.
Addressing the Challenges

• Addressing challenges through set of integrated standards
  – Define a semi-formal methodology to address weakness space coverage
  – Graphically capture claims and evidence (common facts) about a system
  – Graphically capture threat-risk assessment information about a system
  – Automate vulnerability path assessments
  – Specifications for a suite of integrated tools providing end-to-end solution

Tools integration possible only through standards
OMG System & Software Assurance Ecosystem

Operational Mission
CONOPS
Security Policy
Environment
...

Attack Patterns
Software Fault Patterns
(Formalized CWE’s)

C&A Fed by Interoperable Models

Operational Facts
NVDB
S/W Risk Analyzer
TOIF

Sys Risk Manager
SACM
DODAF

Threat Risk Assessment
Structured Assurance Case

Knowledge Discovery
Metamodel: ISO/IEC 19506

Architecture

June 22, 2016
Assurance Claims with Support of ‘Substantial’ Reasoning

• Claims (Conclusion) are assertions put forward for general acceptance.
• Grounds are the justification for claim based on specific Facts, Evidence, or Data about a precise situation that clarify and make good a claim.
• Warrant is the basis of the reasoning from the grounds to the claim is articulated.
  • Toulmin coined the term “warrant” for “substantial argument”.
• Modality are statements indicating the general ways of argument being applied in a particular case and implicitly relied on and whose trustworthiness is well established.
• The basis of the warrant might be questioned, so “Backing” for the warrant may be introduced. Backing might be the validation of the scientific and engineering laws used.

Assurance and Evidence (NIST SP800-160)

- Assurance is best grounded in relevant and credible evidence used to substantiate a claim
  - “the system is acceptably safe / secure”
- An assurance case relate claims and evidence
  - Via structured argumentation and argument patterns
  - Automated via assurance case tools
Claims, Arguments, and Evidence

Claim = assertion to be proven

Argument = how evidence supports claim

Evidence = required documentation
ISO/IEC 15026: Systems & Software Assurance
15026 Part 2: The Assurance Case (Claims-Evidence-Argument)
ISO/IEC 15026: Systems & Software Assurance
15026 Part 2: The Assurance Case (Claims-Evidence-Argument)
Assurance Cases Can Be Large & Composed of Other Assurance Cases

A: Overview of Assurance Case
B: Supplier Practices Reduce Supply Chain Risk
C: Developed/Updated Product is Acceptably Secure
D: Delivered/Updated Product is Acceptably Secure & The Product is Used in a Secure Manner
E: Supplier Has Effective Processes in Place to Support Secure Development

June 22, 2016
The Common Criteria – Top Level

CLAIM
EAL4
[Confidence in Security
because the product has
been] methodically designed,
tested, and reviewed

ARGUMENT
ACM
Configuration Management

ARGUMENT
ACM_AUT
CM Automation

ARGUMENT
ACM_CAP
CM Capabilities

ARGUMENT
ACM_SCP
CM Scope

ARGUMENT
ADO
Delivery and Operation

ARGUMENT
ADO_DEL
Delivery

ARGUMENT
ADO_IGS
Installation, Generation and Start-up

ARGUMENT
ADV
Development

ARGUMENT
ADV_FSP
Development

ARGUMENT
ADV_HLD
High-Level Design

ARGUMENT
ADV_IMP
Implementation

ARGUMENT
ADV_LLD
Low-Level Design

ARGUMENT
ADV_HLD
High-Level Design

ARGUMENT
ADV_IMP
Implementation

ARGUMENT
ADV_LLD
Low-Level Design

ARGUMENT
ADV_RCR
Representation Correspondence

ARGUMENT
AGD
Guidance Documents

ARGUMENT
AGD_ADM
Administrator Guidance

ARGUMENT
AGD_USR
User Guidance

ARGUMENT
AGD
Guidance Documents

ARGUMENT
AGD_ADM
Administrator Guidance

ARGUMENT
AGD_USR
User Guidance

ARGUMENT
ALA
Life Cycle Support

ARGUMENT
ALC_DVS
Development Security

ARGUMENT
ALC_LCD
Life Cycle Definition

ARGUMENT
ALC
Life Cycle Support

ARGUMENT
ALC_DVS
Development Security

ARGUMENT
ALC_LCD
Life Cycle Definition

ARGUMENT
AVA
Vulnerability Assessment

ARGUMENT
AVA_MSU
Misuse

ARGUMENT
AVA_SOF
Strength of TOE Security Functions

ARGUMENT
AVA
Vulnerability Assessment

ARGUMENT
AVA_MSU
Misuse

ARGUMENT
AVA_SOF
Strength of TOE Security Functions

ARGUMENT
ATE
Tests

ARGUMENT
ATE_COV
Coverage

ARGUMENT
ATE_DPT
Depth

ARGUMENT
ATE_FUN
Functional Tests

ARGUMENT
ATE
Tests

ARGUMENT
ATE_COV
Coverage

ARGUMENT
ATE_DPT
Depth

ARGUMENT
ATE_FUN
Functional Tests

ARGUMENT
ATA
Tools and Techniques

ARGUMENT
ATA
Tools and Techniques

ARGUMENT
ATA
Tools and Techniques
CLAIM: 7.0 SCM process is properly established and executed

ARGUMENT: (not explicit)
Satisfactory SCM process requires three characteristics

CLAIM: 8.0 SQA process is properly established and executed

ARGUMENT: (not explicit)
Satisfactory SQA certification liaison process comprises three factors

CLAIM: 2.0 System Aspects are Taken into Account

ARGUMENT: (not explicit)
Certification expects all systems considerations to be addressed

CLAIM: 3.0 Software Life Cycle is properly defined

ARGUMENT: (not explicit)
All three areas are based on “best practices” and have detailed sub-claims

CLAIM: 4.0 Software Planning Process is executed

CLAIM: 5.0 Software Development Process is executed as planned

CLAIM: 6.0 Software Verification [low-level] and Integration Process

ARGUMENT: (not explicit)
Satisfactory verification covers products of all processes

CLAIM: 9.0 Certification Liaison process is properly established & executed

ARGUMENT: (not explicit)
Satisfactory certification liaison process requires three characteristics

CLAIM: 4.0 Software Planning Process is executed

CLAIM: 3.0 Software Life Cycle is properly defined

CLAIM: 2.0 System Aspects are Taken into Account

CLAIM: DO-178B Software Considerations are taken into account

ARGUMENT: (not explicit)
Certification expects all factors be included

CLAIM: 5.0 Software Development Process is executed as planned

CLAIM: 6.0 Software Verification [low-level] and Integration Process

CLAIM: 7.0 SCM process is properly established and executed

CLAIM: 8.0 SQA process is properly established and executed

CLAIM: 9.0 Certification Liaison process is properly established & executed

ARGUMENT: (not explicit)
Satisfactory SQA certification liaison process comprises three factors

ARGUMENT: (not explicit)
Satisfactory SCM process includes six elements

ARGUMENT: (not explicit)
Satisfactory verification covers products of all processes

CLAIM: 2.0 System Aspects are Taken into Account

CLAIM: 3.0 Software Life Cycle is properly defined

CLAIM: 4.0 Software Planning Process is executed

CLAIM: 5.0 Software Development Process is executed as planned

CLAIM: 6.0 Software Verification [low-level] and Integration Process

CLAIM: 7.0 SCM process is properly established and executed

CLAIM: 8.0 SQA process is properly established and executed

CLAIM: 9.0 Certification Liaison process is properly established & executed

ARGUMENT: (not explicit)
Satisfactory SQA certification liaison process comprises three factors

ARGUMENT: (not explicit)
Satisfactory SCM process includes six elements

ARGUMENT: (not explicit)
Satisfactory verification covers products of all processes

CLAIM: DO-178B Software Considerations are taken into account

ARGUMENT: (not explicit)
Certification expects all factors be included
Is the Model Complete?

OV-2 View in DoDAF Model

North Sea Search and Rescue DoDAF Model

Actual Performers Found by Risk Manager
Analysis of Operational Activities
Full Analysis of DoDAF Model

<table>
<thead>
<tr>
<th>SAR Performers</th>
<th>SAR ExchangeElements</th>
<th>SAR Performers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Aircraft Instruction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Life Preserver Instruction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Request for Assistance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Boat Instruction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Medical Advice</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Weather Forecast</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Beacon Instruction</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Reported Location</td>
<td></td>
</tr>
<tr>
<td>Monitoring Node</td>
<td>Track Info</td>
<td>Person in Distress</td>
</tr>
<tr>
<td>Person in Distress</td>
<td>Distress Signal</td>
<td>SAR Concept</td>
</tr>
<tr>
<td>Person in Distress</td>
<td>Distress Signal</td>
<td>Tactical C2 Node</td>
</tr>
<tr>
<td>Person in Distress</td>
<td>Distress Signal</td>
<td>Search Node</td>
</tr>
<tr>
<td>Place of Safety</td>
<td>-</td>
<td>Monitoring Node</td>
</tr>
<tr>
<td>Rescue Node</td>
<td>Medical Condition1</td>
<td>Rescue Node</td>
</tr>
<tr>
<td>Rescue Node</td>
<td>Updated Location</td>
<td>SAR Concept</td>
</tr>
<tr>
<td>SAR Asset Controller</td>
<td>Task1</td>
<td>Search Node</td>
</tr>
<tr>
<td>SAR Asset Controller</td>
<td>Task</td>
<td>SAR Asset Controller</td>
</tr>
<tr>
<td>SAR Concept</td>
<td>-</td>
<td>SAR Asset Controller</td>
</tr>
<tr>
<td>Search Node</td>
<td>Warning Order</td>
<td>Place of Safety</td>
</tr>
<tr>
<td>Search Node</td>
<td>Medical Condition</td>
<td>Rescue Node</td>
</tr>
<tr>
<td>Tactical C2 Node</td>
<td>Control Order1</td>
<td>Search Node</td>
</tr>
<tr>
<td>Tactical C2 Node</td>
<td>Control Order</td>
<td>SAR Asset Controller</td>
</tr>
<tr>
<td>Tactical C2 Node</td>
<td>Request</td>
<td>SAR Asset Controller</td>
</tr>
</tbody>
</table>
OMG System & Software Assurance Ecosystem

Operational Mission
CONOPS
Security Policy Environment
...

Attack Patterns

A&A
Fed by
Interoperable Models

Structured Assurance Case

Threat Risk Assessment

Software Fault Patterns
(Formalized CWE’s)

Knowledge Discovery
Metamodel: ISO/IEC 19506

Architecture

Operational Facts

NVDB

Common Fact Model
(Evidence)

SACM

Sys Risk Manager

DODAF

TOIF

S/W Risk Analyzer

OMG System & Software Assurance Ecosystem

June 22, 2016
Threat Modeling

http://www.omg.org/hot-topics/threat-modeling.htm
OMG System & Software Assurance Ecosystem

Operational Mission
CONOPS
Security Policy
Environment
...

Attack Patterns
NVDB
Operational Facts

A&A
Fed by
Interoperable Models

Threat Risk Assessment
Sys Risk Manager
SACM

Common Fact Model
(Evidence)

KDM Engine
Knowledge Discovery
Metamodel: ISO/IEC 19506

Software Fault Patterns
(Formalized CWE’s)

Architecture

S/W Risk Analyzer
TOF
Facts

Knowledge Discovery
Metamodel: ISO/IEC 19506

June 22, 2016
TOIF Architecture

- **Open Source Flaw / Defect detection tools (SCA)**
  - CPPcheck
  - FindBug
  - JLint
  - RATS
  - Splint
  - Fortify*
  - CodeSonar

  *COTS SCA tools can be adapted

- **TOIF adapters (normalization)**
  - TOIF XMI

- **TOIF analyzer**
  - (unification, correlation and confidence)

- **Fact Oriented Interface**
  - Integrated facts

- **TOIF Open Source**
  - Nov 2012

- **KDM Analytics**
  - Architecture risk analysis report

- **Planned basis for Threat Risk Assessment Metamodel**

**Simplifies Usage for Developers**
- Adapts multiple SCA tools into Common Framework
- Standardizes Output
- Reports Results in OSS Eclipse IDE

**June 22, 2016**
Commercial Tool Coverage (67.2%)
Reported at IEEE Metrocon 2014
Software Risk Analyzer

Compare the Design Information to Implemented Code

June 22, 2016
Threat Risk Analysis of Attack Paths

The architectural component where the buffer overflow is happening.

Threat Risk Analysis discovers attacker has direct access to “Capture Module”

Software Flaw Findings from TOIF
OMG System & Software Assurance Ecosystem

Operational Mission
CONOPS
Security Policy
Environment
...

Implementation
System Artifacts
- Req.
- Use Cases
- Design
- Data Flow Diagrams
- STIGs
...

Software Fault Patterns
(Formalized CWE’s)

Attack Patterns

A&A
Fed by
Interoperable Models

Threat Risk Assessment

Structured Assurance Case

Common Fact Model
(Evidence)

Operational Facts

NVDB

Sys Risk Manager

SACM

DODAF

KDM Engine
Knowledge Discovery
Metamodel: ISO/IEC 19506

Implementation
System Artifacts
- Req.
- Use Cases
- Design
- Data Flow Diagrams
- STIGs
...

Software Fault Patterns
(Formalized CWE’s)

Architecture

June 22, 2016
## Weakness Detection Tool vs. SwA Tool Suite: Different Requirements

<table>
<thead>
<tr>
<th>SwA Key Report Requirements / Weakness Detection Tools</th>
<th>Detected Weakness (W) Set</th>
<th>Path Coverage Breadth per W</th>
<th>Path Coverage Depth per W</th>
<th>Path Coverage Completeness per W</th>
<th>Layered Services</th>
<th>Source code Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sw Assurance Ecosystem</strong></td>
<td>Open, (Rule-based)</td>
<td>Full, transparent</td>
<td>Full, configurable, transparent</td>
<td>Full, transparent</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Source Code Static Analysis</strong></td>
<td>Extensible, usually closed</td>
<td>Can be opportunistic, not transparent</td>
<td>Limited to app layer, can be opportunistic, not transparent</td>
<td>Limited to app layer, no coverage data available</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Binary Static Analysis</strong></td>
<td>With limitations</td>
<td>Can be opportunistic, not transparent</td>
<td>Can be opportunistic, not transparent</td>
<td>Can be opportunistic, not transparent</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td><strong>Penetration Testing</strong></td>
<td>Only vulnerability</td>
<td>Limited by complexity</td>
<td>Limited by complexity</td>
<td>Unknown</td>
<td>Y</td>
<td>N</td>
</tr>
</tbody>
</table>

June 22, 2016
Conclusion

• Structured Assurance Models
  – Bring structured order to chaos
  – Interrelated Claims – Arguments – Evidence between various sources of evidence

• System Risk Manager
  – Analysis of DoDAF model Operation, System, … Views
  – Automated Gap Assessments in Models
  – Threat Risk Assessment capability on DoDAF models

• TOIF and Risk Analyzer tools have demonstrated
  – Significant improvement in Software Flaw and Vulnerability assessments
  – Lower labor costs
  – Significantly lower tool costs

OMG System Assurance Modeling Tools can Reduce Security Engineering Life-cycle costs 20-50%.