A Security Metric Based on Security Arguments

John Knight
University of Virginia
& Dependable Computing LLC

Ben Rodes
University of Virginia

Kim Wasson
Dependable Computing LLC
The Problem

- Is this system *secure*?
- What does this question actually *mean*?

Connectivity

The World Really Wants A Security Metric

The Internet
Existing Security Metrics

- Herrmann (900):

- Checklists and audits

- Conformance to standards:
  - Common Criteria:

73 secure state — state in which the TSF data are consistent and the TSF continues correct enforcement of the SFRs

74 security attribute — property of subjects, users (including external IT products), objects, information, sessions and/or resources that is used in defining the SFRs and whose values are used in enforcing the SFRs

75 security function policy — set of rules describing specific security behaviour enforced by the TSF and expressible as a set of SFRs
Concerns…

- Does conformance with a standard mean my system is secure?
- Do standards apply to my system?
- Do checklists cover all the vulnerabilities in my system?
- Do metrics take my threat models into account?
- Is testing complete in any sense?
- What do “red teams” actually mean?
A Different View

- Stakeholders *know* what security they need:
  - Or at least they should (CC includes this)
- So:
  
  *Define* precisely desired set of security properties

- System development/maintenance:
  
  Engineer with *appropriate* and *suitable* technology
  in system development

- **Real** question:
  
  *Does the implementation achieve the desired security properties fully?*
The Big Picture

- System Concept
  - Functionality Requirements
  - Security Requirements

- System Development
  - Delivered System
  - Implied Claim: System Implements Security Requirements

- Do We Believe This?
Security Case

- **Security Requirements**
- **Claim**: System implements security requirements
- **Rigorous Argument**
- **Security Metric**: Degree of confidence in argument

- **Need To Be Able To Believe** This Argument

- **Contrast**:
  - **Not**: Is security good enough?
  - **But**: Does implementation justify belief in security requirements
Rigorous Security Argument

- Goal structured argument
- Described in GSN
- Explicit documentation of rationale
- Reveals limitations of defense techniques
- Allows explicit scrutiny
Elements of a Security Case

- Security Claim (From Stakeholders)
  - Argument
  - Context
  - Assumptions
  - Justification

- Supporting Evidence (From Developers)
  - Security Metric
  - Degree of Belief in the Argument

- Claim
  - Stating Desired Security Properties

- Explicit Rationale
  - For Belief in Claim
  - Tests
  - Audits
  - Proofs
  - Inspections
  - Other...
Argument Assessment

- Individual ACPs assessed as appropriate:
  - Quantification
  - Expert judgment
- Yields vector of values (or more)

Determine The Metric For A Specific Argument
Confidence Assessment

Confidence Measurement

Assurance Claim Point

Rigorous Argument
Derivation Of Metrics

Annotated Security Argument

Normalization

Put Data Onto Consistent Scale

Synthesize Summary Value

Summarization

Visualization

Possible Displays

Make Available To Stakeholders

< 0.55, 0.965, 0.139 >

< accept, weak accept, strong reject >
Example: “Traffic Light” Metric

Green: Believe
Yellow: Doubt
Red: Serious Doubt

Change Color By:
• Targeted Enhancement
• Restricted Operating Context
Summary

- Security metric highly desirable
- “Measuring” software in some way problematic:
  - What do values mean?
  - What is good enough?
- Proposed approach:
  - Define desired security goal
  - Develop security case including rigorous argument
  - Assess confidence in argument:
    - High confidence suggests security goal met
    - Weaker elements of argument suggest where to repair
- Evaluation:
  - Fragments of arguments initially using argument toolkit
  - Next - experimental application
Questions?

“The 9000 series has a perfect operational record.

Quite honestly, I wouldn’t worry myself about that.”