Costing Software Security
A Systematic Mapping

33rd International Forum on COCOMO® and Software/Systems Cost Modeling

Elaine Venson, USC

October, 2018
CVE Vulnerabilities

Vulnerabilities By Year

https://www.cvedetails.com/browse-by-date.php
Software Vulnerabilities

• Why the problem is growing?
  – Connectivity
    • Connectivity through Internet is increasing
    • People, business and government become more dependent on networked systems
    • Greater risks for poor software security practices
  – Extensibility
    • Nature of extensible systems makes it hard to prevent software vulnerabilities
Software Vulnerabilities

• Why the problem is growing?
  – Complexity
    • Software systems are growing in size and complexity
    • The defect rate tends to go up as the square of code size\(^1\)
    • Other factors contribute – architectural issues, overlay of patches, etc
    • Unsafe programming languages

(1) https://homepage.cs.uri.edu/~thenry/resources/unix_art/ch04s01.html
Complexity Growth

SOFTWARE SIZE (MILLION LINES OF CODE)

Source: NASA, IEEE, Wired, Boeing, Microsoft, Linux Foundation, Ohioh

- Modern High-end car
- Facebook
- Windows Vista
- Large Hadron Collider
- Boeing 787
- Android
- Google Chrome
- Linux Kernel 2.6.0
- Mars Curiosity Rover
- Hubble Space Telescope
- F-22 Raptor
- Space Shuttle

http://qnxauto.blogspot.com/2016/10/
Security

• Security remains largely focused on post-development measures late in the SDLC
  – Vulnerabilities keep growing
• Security is an emergent property of a software system
  – There is no single addition that can make a software secure
• Building secure software is better than protecting bad software
Definitions

• **Cybersecurity, computer security or IT security** is the protection of computer systems from theft of or damage to their hardware, software or electronic data, as well as from disruption or misdirection of the services they provide.¹

• **Application security** means the protection of software after it’s already built. [McGraw, 2004]

• **Software security** is the idea of engineering software that continues working under malicious attack [McGraw, 2004]

Secure Software Development

- Software security must be part of a full life-cycle approach
  - Security requirements (misuse cases, abuser stories)
  - Architectural and design reviews
  - Code reviews
  - Static and dynamic code analysis
  - Risk-driven testing
  - Analysis and auditing of software artifacts

- It is believed that Secure Software Engineering can [Heitzenrater, 2016]:
  - Limit exposure and reduce security incidents
  - Be an investment that decreases overall software expenditure
Problem

Practices to address software security are being increasingly introduced in SDLC. However, current models for software effort estimation do not consider the security factor, causing inaccurate estimates results, which affects the planning of required resources for software project development.
Why a systematic mapping?

- Limited knowledge about how the relationship between software security and cost estimation has been studied.
- Synthesize the knowledge available.
- Find the gaps that indicate the need for further research.
- Identify ‘clusters’ of studies to support the proposition of improvements in current cost models.
Goal

To identify the state-of-the-art in research about the relation between software security practice and development costs/effort.
Method

- Search and identify all relevant material relating Software Security and Software Development Costs
- Follow objective, analytical and repeatable procedures, following Kitchenham et al [2016]
- A secondary study -> generate outcomes by aggregating material from primary studies
Mechanism of the Search

Tittle:
- Research question 1
- Research question 2
- Research question 3

Manual Search → Identify relevant venues → Quasi-gold standard

Objectively:
- Derive search terms

Automated Search ← Complement

Subjectively:
- Evaluate

Relevant Studies

## Venues of Search

<table>
<thead>
<tr>
<th>Manual Search</th>
<th>Automated Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IEEE Transactions on Software Engineering (TSE)</td>
<td>• IEEE Digital Library</td>
</tr>
<tr>
<td>• ACM Transactions on Software Engineering Methodology (TOSEM)</td>
<td>• ACM Digital Library</td>
</tr>
<tr>
<td>• Empirical Software Engineering Journal</td>
<td>• SpringerLink</td>
</tr>
<tr>
<td>• Journal of Systems and Software</td>
<td>• Scopus</td>
</tr>
<tr>
<td>• Information and Software Technology</td>
<td>• Web of Science</td>
</tr>
<tr>
<td>• Proceedings of the International Conference on Software Engineering (ICSE)</td>
<td></td>
</tr>
<tr>
<td>• Empirical Software Engineering and Metrics Conference (ESEM)</td>
<td></td>
</tr>
<tr>
<td>• Workshop on Software Engineering for Secure Systems (SESS)</td>
<td></td>
</tr>
<tr>
<td>• Software and Systems Modeling</td>
<td></td>
</tr>
</tbody>
</table>
## Primary Study Selection

<table>
<thead>
<tr>
<th>Inclusion Criteria</th>
<th>Exclusion Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IC1-Paper proposes and estimation model that includes security as a driver for</td>
<td>• reactive approach to software security issues</td>
</tr>
<tr>
<td>effort/cost.</td>
<td>• research about software safety</td>
</tr>
<tr>
<td>• IC2-Paper investigates how to predict effort for specific tasks related to</td>
<td>• not presented in English</td>
</tr>
<tr>
<td>software security development.</td>
<td>• not accessible in full-text</td>
</tr>
<tr>
<td>• IC3-Paper describes a method for secure development and add information or</td>
<td>• book or gray literature</td>
</tr>
<tr>
<td>make considerations about effort or cost-effectiveness of the method.</td>
<td>• tutorial, workshop or poster summary</td>
</tr>
<tr>
<td>• IC4-Paper describes research on software estimation in general and describes</td>
<td>• study is duplicated</td>
</tr>
<tr>
<td>security issues in estimating.</td>
<td></td>
</tr>
<tr>
<td>• IC5-Paper describes research on security economics related to software</td>
<td></td>
</tr>
<tr>
<td>development.</td>
<td></td>
</tr>
<tr>
<td>• IC6-Paper published between 2000 up to and including 2017.</td>
<td></td>
</tr>
</tbody>
</table>
## Results from Search

### Manual Search

<table>
<thead>
<tr>
<th>Venue</th>
<th># Papers</th>
<th># 1st Filter</th>
<th># Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSE</td>
<td>1762</td>
<td>30</td>
<td>2</td>
</tr>
<tr>
<td>TOSEM</td>
<td>407</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>EmSE</td>
<td>705</td>
<td>16</td>
<td>2</td>
</tr>
<tr>
<td>JSS</td>
<td>3014</td>
<td>64</td>
<td>1</td>
</tr>
<tr>
<td>IST</td>
<td>1803</td>
<td>80</td>
<td>1</td>
</tr>
<tr>
<td>ICSE</td>
<td>2538</td>
<td>52</td>
<td>1</td>
</tr>
<tr>
<td>ESEM</td>
<td>703</td>
<td>56</td>
<td>1</td>
</tr>
<tr>
<td>SESS</td>
<td>72</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>SoSyM</td>
<td>653</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>11657</strong></td>
<td><strong>324</strong></td>
<td><strong>9</strong></td>
</tr>
</tbody>
</table>

### Snowballing

<table>
<thead>
<tr>
<th>Iteration</th>
<th># Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>37</strong></td>
</tr>
</tbody>
</table>
Results from Search

Manual Search
• 9 papers

Snowballing
• 37 papers
• 5 iterations

Quasi-Gold Standard
• 46 papers

Automatic Search
• 11 papers
• 86% recall
• Low precision
• 10 strings tested

Total: 57 papers
Results
#Papers per Channel

- VARIOUS: 43 papers
- ITSIM: 2 papers
- IEEE TRANSACTIONS ON SOFTWARE ENGINEERING: 2 papers
- IEEE SECURITY PRIVACY: 2 papers
- ESSOS: 2 papers
- EMPIR SOFTWARE ENG: 2 papers
- ARES: 4 papers
#Papers according to Category

<table>
<thead>
<tr>
<th>Category</th>
<th>Papers</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC1 - SECURE SOFTWARE ESTIMATION MODEL</td>
<td>7</td>
</tr>
<tr>
<td>IC2 - ESTIMATION OF SECURITY ACTIVITY</td>
<td>8</td>
</tr>
<tr>
<td>IC3 - SOFTWARE SECURITY PRACTICE ANALYSIS</td>
<td>27</td>
</tr>
<tr>
<td>IC4 - SECURITY CONSIDERATIONS ON EFFORT</td>
<td>4</td>
</tr>
<tr>
<td>IC5 - SOFTWARE SECURITY ECONOMICS</td>
<td>11</td>
</tr>
</tbody>
</table>
#Papers according to Category

- **IC1 - SECURE SOFTWARE ESTIMATION MODEL**: 7 papers
- **IC2 - ESTIMATION OF SECURITY ACTIVITY**: 8 papers
- **IC3 - SOFTWARE SECURITY PRACTICE ANALYSIS**: 27 papers
- **IC4 - SECURITY CONSIDERATIONS ON EFFORT**: 4 papers
- **IC5 - SOFTWARE SECURITY ECONOMICS**: 11 papers
### Papers on Cost Model for Developing Secure Software


Estimating Software Security

• Existing models:
  – Few were validated with representative datasets
  – Based on Common Criteria (CC) standard
  – COCOMO and FPA derivations

• Challenges:
  – Large variation in existing security standards
  – Lack of historical data to validate methods
  – Proposed methods or models for estimation of secure software systems not validated
# Papers according to Category

- **IC1 - Secure Software Estimation Model**: 7 papers
- **IC2 - Estimation of Security Activity**: 8 papers
- **IC3 - Software Security Practice Analysis**: 27 papers
- **IC4 - Security Considerations on Effort**: 4 papers
- **IC5 - Software Security Economics**: 11 papers
Software Security Practice

#Papers by Subcategory

- Security Requirements: 1
- Quantification of Software Security: 1
- Security Practices in Software Development: 2
- Security Development Method: 3
- Security in Agile Processes: 6
- Vulnerability Prediction/Detection: 7
- Countermeasure Identification/Prioritization: 7
# Papers according to Category

IC1 - Secure Software Estimation Model
- 7 Papers

IC2 - Estimation of Security Activity
- 8 Papers

IC3 - Software Security Practice Analysis
- 27 Papers

IC4 - Security Considerations on Effort
- 4 Papers

IC5 - Software Security Economics
- 11 Papers
Software Security Economics

- Software security is costly
  - Impede best practices to be implemented
- Justifying software security expenditure:
  - Insurance (reduction of potential risks)?
  - Or investment (benefits of early expenditure)?
- How much security is enough?
  - Need of economically-informed decisions
  - Return on Secure Software Process (ROSSP)
  - Unification of pre- and post-security investments
# Software Security Economics

<table>
<thead>
<tr>
<th>Paper</th>
<th>Title</th>
<th>Authors</th>
<th>Pub. Date</th>
<th>Pub. Type</th>
<th>Pub. Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>P6</td>
<td>Security attribute evaluation method: a cost-benefit approach</td>
<td>S. A. Butler</td>
<td>2002</td>
<td>Conference</td>
<td>ICSE</td>
</tr>
<tr>
<td>P51</td>
<td>Security Requirements Prioritization Based on Threat Modeling and Valuation Graph</td>
<td>Keun-Young Park, Sang-Guun Yoo, and Juho Kim</td>
<td>2011</td>
<td>Conference</td>
<td>ICHIT</td>
</tr>
<tr>
<td>P56</td>
<td>A jump-diffusion approach to modelling software security investment</td>
<td>J. Zheng; J. Wan; Y. Ren; H. Guo</td>
<td>2012</td>
<td>Conference</td>
<td>BIFE</td>
</tr>
<tr>
<td>P55</td>
<td>Secure It Now or Secure It Later: The Benefits of Addressing Cyber-Security from the Outset</td>
<td>Olama, MM; Nutaro, J</td>
<td>2013</td>
<td>Conference</td>
<td>SPIE</td>
</tr>
<tr>
<td>P17</td>
<td>A case for the economics of secure software development</td>
<td>Chad Heitzenrater, Andrew Simpson</td>
<td>2016</td>
<td>Conference</td>
<td>NSPW</td>
</tr>
<tr>
<td>P32</td>
<td>The Impact of Security by Design on the Success of Open Source Software</td>
<td>Golriz Chehrazi, Irina Heimbach, Oliver Hinz</td>
<td>2016</td>
<td>Conference</td>
<td>ECIS</td>
</tr>
<tr>
<td>P33</td>
<td>Misuse, Abuse, and Reuse: Economic utility functions for characterising security requirements</td>
<td>Chad Heitzenrater, Andrew Simpson</td>
<td>2016</td>
<td>Conference</td>
<td>ARES</td>
</tr>
<tr>
<td>P34</td>
<td>Software Security Investment: The Right Amount of a Good Thing</td>
<td>Chad Heitzenrater ; Andrew Simpson</td>
<td>2016</td>
<td>Conference</td>
<td>SecDev</td>
</tr>
</tbody>
</table>
Security Investment over SDLC

Current State

Not a favorable allocation

More likely to be effective

Most appropriate to SSE community

(a) Notional depiction showing the bulk of security investment occurring late in the lifecycle (e.g., post-deployment).

(b) Notional depiction of a constant security budget $B_s$ as a fixed proportion of the per-phase budget $B$ (here, as 20%) over the course of a project.

(c) Notional depiction of security investment that starts early in the lifecycle and slowly increases.

(d) Notional depiction of security investment at the early stages, leading to reduced investment in later phases.

Return on Secure Software Process (ROSSP)

Next Steps

• Extract sources of cost and related cost drivers
• Understand how theories behind software security economics can impact software security estimation
• Conduct causal learning study with factors identified.
References

• https://www.commoncriteriaportal.org
Backup Slides
Software Security

• Software is a **central and critical aspect** of the computer security problem
• Many issues faced in computer security today are **rooted in our approach** to developing software and systems
• Software defects have **security ramifications**
• Malicious intruders can hack into systems by **exploiting** software defects
Research Questions

1. Which papers describe research related to estimation/measuring the effort or cost of security in software development?

2. What are the major sources of costs in developing secure software?

3. What approaches have been used to estimate the costs of security in software development projects?

4. Which software security standards and formal assessments have been used in estimation models?
Research Questions (cont)

5. Which and how are the factors that have effect on cost (cost drivers) affected by security requirements?

6. What issues have been observed when measuring or estimating costs for secure software development?

7. Which data sets have been used to analyze the cost of secure software?
Systematic Review Process

Source: Kitchenham, B.A., Budgen, D., Brereton, P.: Evidence-Based Software Engineering and Systematic Reviews (adapted).

©USC-CSSE 38
Context of a Systematic Review

Primary Studies
- case study
- laboratory experiment

Systematic Review

- provides input to
- creates

Objective summary of evidence about a technology, practice, etc.

- influences
- provides inputs to

Practice

Policies

Standards

Source: Kitchenham, B.A., Budgen, D., Brereton, P.: Evidence-Based Software Engineering and Systematic Reviews (adapted).

11/05/18
Steps in the Systematic Mapping

Select Studies
- Inclusion/exclusion criteria

Assess Quality
- Quality checklist
- Rigor and relevance of primary studies

Extract Data
- Extraction form
- Evidence to support the RQ

Synthesize
- Narrative synthesis
- Thematic Analysis

Document

Validation Activities
(check decisions)

11/05/18
Completeness: Sensitivity & Precision

**Sensitivity**

\[
\text{Sensitivity} = \frac{\text{Number of relevant studies retrieved}}{\text{Total number of relevant studies}} \times 100\%
\]

**Precision**

\[
\text{Precision} = \frac{\text{Number of relevant studies retrieved}}{\text{Number of studies retrieved}} \times 100\%
\]

Costs to find and fix defects

- Finding and fixing a **severe** software problem after delivery is often **100 times more expensive** than finding and fixing it during the requirements and design phase.
- Finding and fixing **non-severe** software defects after delivery is **about twice as expensive** as finding these defects pre-delivery.

Common Criteria

- Provides a **common set of requirements** for the security functionality of IT products and for assurance measures applied to these IT products during a security evaluation.

- Establishes a **level of confidence** that the security functionality of these IT products and the assurance measures applied to these IT products meet these requirements.

- Is useful as a **guide for the development, evaluation and/or procurement of IT products** with security functionality.
Common Criteria EALs

- Seven Evaluation Assurance Levels (EALs)
  - EAL1: Functionally Tested
  - EAL2: Structurally Tested
  - EAL3: Methodically Tested and Checked
  - EAL4: Methodically Designed, Tested, and Reviewed
  - EAL5: Semi-Formally Designed and Tested
  - EAL6: Semi-Formally Verified Design and Tested
  - EAL7: Formally Verified Design and Tested
Cost Security for COTS


- Security risks of integrating COTs in critical operations.
- Optional security cost driver to be used with COCOMO II®, based on Common Criteria.
- Initial expert-based cost driver values obtained through Delphi exercise:
  - Security adds a nominal 50% effort when of COTS.
  - Range of impact between 10% and 95% on effort.
  - And between 5% and 43% on schedule.
- Next step was to statistically validate model’s accuracy.
COSECMO


• COSECMO is an extension of COCOMO II® model, with the addition of:
  – A new cost driver “SECU”.
  – Guidance on estimating the size of security functions.
  – Guidance on setting levels of existing COCOMO drivers that are affected by security.
COSECMO

- **COSECMO extensions adds:**
  - Effort to develop the additional software needed to implement the security functions
  - Effort to assure that the system is secure

- **Formulas:**

  PM = Effort (Assured EC) = Effort (EC) + Effort (Internal Assurance)

  Effort (Internal Assurance) = Effort (EC) * %Effort (AL)

  %Effort (EAL) = %Effort _3 * SECU _{EAL - 3} for EAL >= 3

  = 0 for EAL <3

  - Effort (EC) is effort in PM for Elaboration and Construction phases
  - Effort (Internal Assurance) is additional developer assurance effort to verify that the system is secure
  - AL is Assurance Level
  - EAL is Evaluation Assurance Level
Extended FPA with Security Estimation


• Extended FPA [Abdullah, 2010], proposes adding Security as a new factor in the General Systems Characteristics

• 48 security characteristics grouped into 10 security aspects are evaluated to compute the degree of influence for Security

• This value is added to the original formula:

\[ VAF = 0.65 + [(TDI + \text{Security}) \times 0.01] \]
MND-SCEMP


- **MND-SCEMP** – Ministry of National Defense-Software Cost Estimation Modeling Process
- An extension of COCOMO II® cost model with adaptations in the modeling methodology
- The model was developed to reflect **specific domain characteristics** of the Korean weapon system development environment
- Cost drivers are classified in:
  - General cost factors (from COCOMO II®)
  - Domain-specific cost factors (from defense domain)
MND-SCEMP

- Domain-specific cost drivers include:
  - Security
  - Interoperability
  - Simultaneity
  - Emulator
  - Precedentdness

- Security is the higher value with 1.875 for the High rating

- The model was calibrated with 73 data-points from weapon system projects in Korea