

# Stakeholder Value Driven Threat Modeling for Off The Shelf Based Systems

Yue Chen

Advisor: Dr. Barry W. Boehm

Center for Systems and Software Engineering

University of Southern California

Los Angeles, CA, 90089-0781, USA

{yuec, boehm}@usc.edu

## Abstract

*This paper abstract summarizes the Threat Modeling method based on Attacking Path Analysis (T-MAP) which quantifies and prioritizes security threats by calculating the total severity weights of relevant attacking paths for Commercial Off The Shelf (COTS) based systems. Compared to existing approaches, T-MAP is dynamic and sensitive to system stakeholder value priorities and IT environment. It distills the technical details of thousands of relevant software vulnerabilities into management-friendly numbers at a high-level. In its initial usage in a large IT organization, T-MAP has demonstrated significant strength in COTS vulnerability prioritizing and estimating security investment effectiveness, as well as COTS security assessment in early project life-cycle. Furthermore, a software tool has been developed to automate the T-MAP.*

## 1. Problem Statement

As the trend of the usage of third party Commercial-Off-The-Shelf (COTS) and open source software continuously increases [34], COTS security has become a major concern for many organizations whose daily business heavily relies upon a healthy IT infrastructure [18]. But, according to the 2006 CSI/FBI computer criminal survey, 47% of the surveyed organizations spent only equal or less than 2% of their IT budget in security [14]. Often, competing with limited IT resources and the fast changing internet threats, the ability to prioritize security vulnerabilities and address them efficiently has become a critical success factor for every security manager.

As known, the security impacts of vulnerabilities can be specified in terms of Confidentiality, Integrity, and Availability (CIA) [27]. These security attributes can have very different business indications under different application context. Unfortunately, most current leading vulnerability rating systems by CERT, ISS, Microsoft, NIST, and Symantec is value neutral, static, and treat CIA equally. To date, it is still very difficult to prioritize security vulnerabilities efficiently with quantitative evidence because of lack of effective metrics, lack of historical data, and the complex and sensitive nature of security [7]. These difficulties can result in (1) inefficient allocation of security resources; (2) making inconsistent security investment decisions based on individual's experience, judgment, and best knowledge; (3) selecting

COTS products without evaluating how and how much the associated vulnerabilities can impact organizational stakeholder values, and take security "as is" reactively after the system is built.

We propose to (1) establish a framework that is sensitive to stakeholder value propositions to dynamically prioritize COTS vulnerabilities and model security threat; (2) grow a comprehensive vulnerability database which integrates the published vulnerability information from multiple authority sources such as the Symantec/BugTraq, NIST/NVD, CERT, Microsoft, and FrSIRT to support T-MAP analysis; (3) automate T-MAP vulnerability evaluation based on a XML framework that abstracts the process.

## 2. Related Work

Our work is primarily relevant to the research areas of COTS based system security, Value Based Software Engineering, and Security Economics.

To date, third party COTS and open source vulnerabilities have been published near real-time and ranked by authority organizations such as CERT, NIST, Symantec/BugTraq, Microsoft, SANS, MITRE, ISS, OSVDB and FrSIRT, etc. [11, 19, 21, 22, 23, 28, 31] While the COTS vulnerabilities are assessed in a promising technical depth, unfortunately, most existing leading approaches are stakeholder value neutral, static, and treat CIA equally. Unavoidably, the rankings can be misleading in some situations. For example, in the NIST Common Vulnerability Scoring System (CVSS), the vulnerability that can only compromise availability has a max rating of 3.3(low) out of 10 [20] (i.e., the CVE-2006-3468). But for many production servers, availability can be mission-critical. In this case, the rating largely missed its initial goal to help prioritize vulnerability.

We propose applying Value Based Software Engineering philosophy and principles to COTS vulnerability evaluation [2, 4, 5] to address these challenges. Fundamentally, successful security should be achieved at a level that makes all key stakeholders winners [4]. The stakeholder utility functions can be identified through Win-Win Negotiation [6] and prioritized with Figure of Merit and Analytical Hierarchy Process (AHP) [2, 6, 29]. The value-security dependencies can be analyzed through Result Chain [30], attack tree [29], data flow of organizational IT operations [16], and the system use/abuse cases [33].

As known, cost-benefit analyses can provide a sound basis for security investment decision making [13]. In

---

\* Manuscript submitted for exclusive review by the 29<sup>th</sup> International Conference on Software Engineering, Dec. 11, 2006.  
All rights are reserved by co-authors.

previous works, Gordon and Loeb presented an quantitative economic model to determine the optimal amount to invest in security to protect a given set of information [12]; Hoo proposed a risk management approach to answer the question “how much security is enough” [15]; Butler demonstrated using the *multiple-attribute risk assessment* in SAEM to reason the cost-benefit of security investments [7]; Cavusoglu et al proposed a quantitative model based on game theory to evaluate security investments [8]. Along with the classic IT risk management methodologies [27, 32], these works established a sound conceptual framework to reason the cost-effectiveness of security investment. However, though, most of the work is still at a very high-level and lack of fine-grained consideration to specific COTS system vulnerability. In addition, the accurate value of many of the parameters used in these models such as probabilities, frequencies and size of loss are usually very difficult to estimate even for experienced security manager.

### 3. Research Hypothesis

**Hypothesis #1:** A framework can be devised to perform fine-grained COTS vulnerability comparison and evaluation for given COTS Based System (CBS), such that given two otherwise *technically identical* COTS vulnerabilities  $V_x$  and  $V_y$ , except for their breach type in terms of Confidentiality, Integrity, and Availability, if the breach type caused by  $V_x$  is more critical than  $V_y$ , in terms of the stakeholder utility priority (measured by Figure of Merit score), then the T-MAP generated severity value for  $V_x$ , represented by  $W(V_x)$ , will be not less than  $W(V_y)$ .

Two COTS vulnerabilities are defined as *technically identical* if their major value neutral characteristics are similar, such as (but may not limit to) (1) the impacts to software and hardware; (2) if the vulnerability can be exploited remotely or locally; (3) if the exploiting the vulnerability require the attacker to be authenticated on the victim system; (4) if the successful exploit of the vulnerability involves victim activities such as opening email attachment; (5) if official/temporal patch or advisory are available to the vulnerability; (6) accessibility to attackers; etc. This working definition is based on the emerging NIST standard CVSS [20].

**Hypothesis #2\*:** For given COTS based system whose confidentiality, integrity and availability have different priorities to the stakeholder values, the *Inaccuracy* of T-MAP results, measured by the ratio of *the number of clashes between vulnerability priorities and stakeholder value priorities and the total number of comparisons*, will not make any difference comparing to the existing stakeholder value-neutral approaches.

**Metrics** A *clash* is defined as: for two *technically identical* vulnerabilities  $V_x$  and  $V_y$ , given the stakeholder value impact severity of  $V_x$  is higher than  $V_y$ , if a prioritizing system assign  $V_x$  equal or less priority than  $V_y$ , it is counted as a *clash* for this prioritizing system.

\* Hypothesis #2 is a null hypothesis and we work toward disprove it through empirical case studies.

### 4. Proposed Solution

We liken measuring the COTS system security to defending a castle:

**Castle Defense Analog:** *measure the security of a castle by the value of treasures in the castle, the number of holes on the walls, as well as the size of the holes.*

In brief, T-MAP involves the following steps: **Step 1:** identify key stakeholders and value propositions (the treasures in the castle); **Step 2:** establish a set of security evaluation criteria based on stakeholder value propositions; **Step 3:** enumerate and analyze attack paths based on a comprehensive COTS vulnerability database containing 23,620 vulnerability information (the holes); **Step 4:** evaluate the severity of each scenario in terms of numeric ratings against the evaluation criteria established in Step 2 (the size of the holes); **Step 5:** the security threat of each vulnerability is quantified with the total severity ratings of all attack paths that are relevant to this vulnerability; **Step 6** system total threat is quantified with the total severity ratings of all attack paths. Step 3~6 are tool automated.

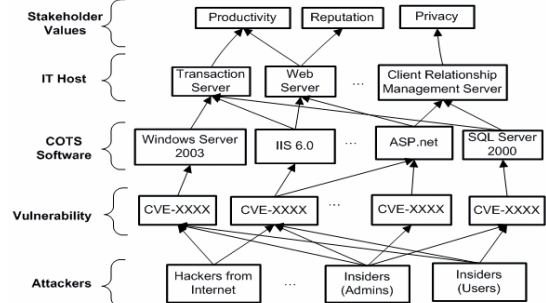


Figure 1 Structured Attack Graph

A formal framework has been devised to enumerate and evaluate attack scenarios based on attack graph analysis. As illustrated in Figure 1, we propose a *Structured Attack Path* approach that incorporates the Schneier’s attack tree and the classic IT risk management framework of attacker, asset, vulnerability, and impact to enumerate the sketchy of attack scenarios [26, 27]. The new attack tree nodes are structured into five layers corresponding to each layer in Figure 1: the first layer nodes represent stakeholder values, for example, productivity, privacy, or reputation. The second layer nodes represent IT hosts that the stakeholder values rely upon. The third layer nodes represent the COTS software that is installed on the IT hosts. The fourth layer nodes represent software vulnerabilities that reside in the COTS software. The fifth layer nodes represent potential attackers, for example, insiders, external hackers, etc.

Clearly, each Attack Path in the Structured Attack Graph represents a uniformed scenario description on how attackers can compromise stakeholder/values through exploiting COTS vulnerabilities in the IT infrastructure. In order to evaluate scenario severity, T-MAP models Attack Path in UML with a set of threat-relevant attributes. The attributes are classified into three categories: *value-impact-relevant*, *exploitability-relevant*, and *descriptive*. Each of the attributes (except for descriptive) is assigned a numeric rating between 0~1: the

value-impact-relevant attributes are rated by evaluating the vulnerability impact in terms of confidentiality, integrity or availability against stakeholder value criteria by using the AHP method [9, 11, 35]; the exploitability-relevant attributes are rated based on the CVSS [20] with several extended attributes[9].

Furthermore, the T-MAP weighting system is established upon the following definitions:

**Def. 1 Weight of Attack Path**

For given Attack Path  $P$ , define:

$$Weight(P) = \prod_i Rating(P.Attribute_i)$$

Where  $P.Attribute_i$  enumerates once each of the Exploitability-Relevant and Value-Relevant attributes of  $P$ .

**Rational:** Analog to the classic risk calculation formula:  
Risk = Probability \* Size of Loss

**Def. 2 Total Threat**

For given Structured Attack Graph  $G$ , define:

$$TotalThreat(G) = \sum_i Weight(AttackPath_i),$$

where  $i$  varies from 1 to the total number of attacking paths of  $G$  and  $AttackPath_i$  represents the  $i$ -th Attack Path of  $G$ .

**Rational:** The Castle Defense Analog

**Def. 3 ThreatKey of COTS Vulnerability**

For a given node  $N$  in a Structured Attack Graph  $G$ , define:

$$ThreatKey(N) = \sum_i Weight(AttackPath_i),$$

where  $i$  varies from 1 to the total number of attacking paths that go through node  $N$ , and  $AttackPath_i$  enumerates all the Attack Paths that go through  $N$ .

**Rational:** The more Attack Path associated with vulnerability, the more severe the vulnerability is.

**Def. 4 Effectiveness of Security Practices**

For a given security practice  $SP$ ,  
 $Effectiveness(SP) = 1 - TotalThreat(AfterSP) / TotalThreat(BeforeSP)$

**Rational:** The effect of a security practice can be simulated by removing the corresponding attack paths and nodes that this security practice can suppress from the graph. For example, the effect of vulnerability patching can be simulated by removing all Attacking Paths that have vulnerability patches available from the Attacking Path set that is before applying patches.

Obviously, the T-MAP weighting system assigns higher *ThreatKey* values to those vulnerability that (1) has higher stakeholder value impacts; (2) impacts more IT hosts and COTS products; (3) are easier to exploit; (4) has more exposure to attackers. Furthermore, we have developed an  $O(|V|+|E|)$  algorithm to calculate the vulnerability *ThreatKey* where  $|V|$  and  $|E|$  is the number of nodes and edges in a *Structured Attack Graph*, respectively.

In the USC-ITS case study [9], the T-MAP generated similar priorities to the one ranked by the security manager manually with a high correlation R square value of 0.86, as illustrated in Figure 2. The result indicates that at least in this case study T-MAP considerably well captured the

human perceptions on stakeholder values through its method steps.

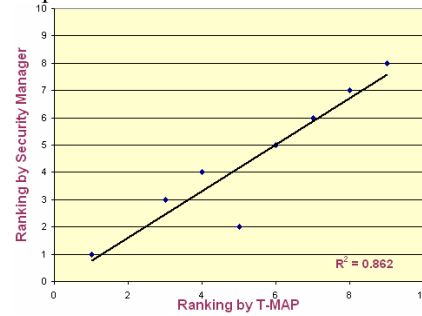


Figure 2 Human-T-MAP Vulnerability Rankings Comparison

Assuming the vulnerabilities that have higher *ThreatKey* values are fixed first, given the average cost of applying a security patch is known, we plotted the production function in terms of *Effectiveness* vs. the number of patches to apply with and without firewall for our case study. This information can be used to answer the question “for a certain amount of security budget, which plan is better.”

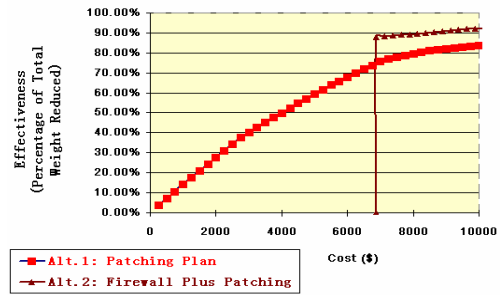


Figure 3 Optimal Security Practice Plan

**5. Expected Contributions**

In our pilot case study, T-MAP demonstrated significant strength in prioritizing vulnerability dynamically based on stakeholder value context and estimating the effectiveness of security practices. It distills the technical details of thousands of published software vulnerabilities into executive-friendly numbers at a high-level. Furthermore, this process can be automated by software tool, which greatly reduces the necessary human effort involved.

Thus, even for large organizations, there is some hope that it can help (1) security administrators identifying key vulnerability based on stakeholder value preferences; (2) executives using cost-effectiveness analyses for their security practices and investments; (3) IT system architects assess COTS security in early project life-cycle, avoiding take security maintenance “as is” after the system is built.

**6. Current Progress**

To date, we have (1) established the initial T-MAP framework; (2) populated a comprehensive COTS vulnerability database that contains 23,620 COTS vulnerability information; (3) developed a software tool automatically crawl and collect vulnerability information from authority security sources including NVD, Symantec/BugTraq, Microsoft, CERT, FrSIRT, and SANS; (4) developed a software tool that automates the T-MAP

framework; (5) conducted two pilot case studies at the USC ITS that is in charge of the USC campus IT infrastructure; (6) applied T-MAP in two security critical CSSE eServices projects to help system architects select COTS alternatives based on the security/value context. The initial results are very positive. More details are presented in section 8.

## 7. Research Methods

Our research is carried out through spiral research increments. In each spiral, the common interests among researchers and practitioners (the USC-ITS, which is in charge of the campus IT infrastructure) are identified and prioritized; related literature studies are conducted; conceptual designs are established and reviewed; related tools are implemented; then the solution and tool are evaluated empirically in real life projects and case studies.

## 8. Evaluation Plan and Initial Results

Hypothesis #1 will be tested by the attack path severity weight calculating algorithm defined in T-MAP. In addition, case studies will be conducted on USC ITS and CSSE eService projects to demonstrate that under otherwise similar conditions, the vulnerabilities have more stakeholder value importance (measured by the Figure of Merit score) are assigned higher priorities than those have less stakeholder value importance.

Hypothesis #2 will be validated empirically through at least three representative case studies from USC-ITS, CSSE eServices, and CSSE affiliates real life projects wherein stakeholder values have clear dependencies and priorities over different security scenarios. The number of *clashes* between vulnerability priorities by rating system and the stakeholder value priorities will be counted for T-MAP as well as other state of the art approaches (i.e. CVSS, etc). The hypothesis #2 will be proved or disproved empirically through statistical t-test.

**Threats to Validation** is summarized in Table 1 as follows:

**Table 1** Threat to Validation

Threat to Validation	Mitigation
Using security expert's vulnerability ranking as "truth" to determine clashes	Conduct case studies with experienced security managers
The number of case studies can be conducted is limited, also the number of vulnerability that a security manager can prioritize manually is limited	Explore more case studies with CSSE affiliates and other possible sources in the CSSE Annual Research Review (March, 2007)
Need for comprehensive vulnerability database to generate meaningful output	Develop automated vulnerability information crawling/collecting engine

**Table 2** Initial Result on Clash Counting

	Number of Clashes	Inaccuracy
T-MAP	2	0.071
CVSS	6	0.214
ISS	9	0.321

**Initial Results** For the pilot case study we conducted at the USC-ITS, the security prioritized 8 vulnerabilities which translates to a total number of 1+2+ ... +7=28 comparisons. The clash counting results for T-MAP and other leading security ranking approaches are summarized

in Table 2. For the pilot case study, the result shows T-MAP significantly out-performed the other two leading approaches. Our clients commented that T-MAP was, "a valuable way of quantifying the very difficult tradeoffs that we have to make everyday."

## 9. References

- [1] R. Baldwin. Rule based analysis of computer security. Technical Report TR-401, MIT LCS Lab, 1988
- [2] B. Boehm, Software Engineering Economics, Prentice Hall PTR, ISBN 0-13-822122-7, Pp223-242, 1981
- [3] V. Basili and B. Boehm, "COTS Based System Top 10 List", Computer, Vol.34, no. 5, 2001, pp.91-93
- [4] B. Boehm and A. Jain, "An Initial Theory of Value-Based Software Engineering" in S. Biffl, A. Aurum, B. Boehm, H. Erdogmus, P. Gruenbacher (eds.), Value-Based Software Engineering, Springer Verlag, 2005.
- [5] B. Boehm, K. Sullivan, Software Economics: A Roadmap, The Future of Software Engineering, ACM 2000, pp 319-343
- [6] L. D. Bodin, L. A. Gordon, M. P. Loeb, Evaluating Information Security Investment Using the Analytic Hierarchy Process, Communications of The ACM, February 2005
- [7] S. A. Butler, Software evaluation: Security attribute evaluation method: a cost-benefit approach, Proceedings of the 24th International Conference on Software Engineering, May 2002
- [8] H. Cavusoglu, B. Mishra, S. Raghunathan, A model for evaluating IT security investments, Communications of the ACM, July 2004
- [9] Y. Chen, B. Boehm, L. Sheppard, Measuring Security Investment Benefit for COTS Based Systems, CSSE Technical Reports #2006-609, 2006
- [10] <http://cve.mitre.org/> (current 11/2006)
- [11] FiSIRT Security Advisories, <http://www.frsirt.org/english>
- [12] Gordon, L., and Loeb, M. The economics of information security investment. *ACM Trans. Inf. Syst. Sec.* 5, 4 (2002), 438-457
- [13] L. A. Gordon, M. P. Loeb, Budgeting process for information security expenditures, Communications of The ACM, January 2006
- [14] L.A. Gordon, M.P. Loeb, W. Lucyshyn, R. Richardson, 2006 CSI/FBI Computer Crime and Security Survey, GoCSI.com, 2006
- [15] Hoo, K.J.S. How much is enough? A risk management approach to computer security. Ph.D. Dissertation, Stanford University, 2000.
- [16] Michael Howard, David LeBlanc, Writing Secure Code, Microsoft Press, 2002, ISBN 0-7356-1722-8, Chapter 4, pp 69-124
- [17] ISO IS 15408, The Common Criteria for Information Technology Security Evaluation (CC) version 2.1, 1999
- [18] R. A. Martin, Managing Vulnerabilities in Your COTS Systems Using An Industry Standards Effort, IEEE, 2002
- [19] Microsoft Security Bulletin, Microsoft Corporation, <http://www.microsoft.com/technet/security/bulletin/> (current 11/2006)
- [20] M. Schiffman, Common Vulnerability Scoring System (CVSS), <http://www.first.org/cvss/> (current 11/2006)
- [21] National Vulnerability Database, NIST, <http://nvd.nist.gov/>
- [22] Open Vulnerability Assess Language, MITRE Corporation, <http://oval.mitre.org/oval/>, (current 11/2006)
- [23] <http://osvdb.org/> (current 11/2006)
- [24] D. Port and Z.H. Chen, "Assessing COTS Assessment: How much is Enough?" Proc. 3<sup>rd</sup> Int'l Conf. COTS Based Software Systems (ICCBSS 04, LNCS 2959, Springer-Verlag, 2004, pp 183-198
- [25] D. Reifer et al., "COTS Based Systems: Twelve Lessons Learned", Proc. 4<sup>rd</sup> Int'l Conf. COTS-Based Software Systems (ICCBSS 04), LNCS 2959, Springer-Verlag, 2004, pp.137-145
- [26] B. Schneier. Attack trees: Modeling security threats. Dr. Dobb's Journal, December 1999
- [27] G. Stoneburner, A. Goguen, A. Feringa, Risk Management Guide for IT Systems, NIST Special Publication 800-30, 2002
- [28] <http://www.securityfocus.com/> (current 11/2006)
- [29] T.L. Saaty, *The Analytic Hierarchy Process*. McGraw-Hill, NY, 1980.
- [30] Thorp, J., DMR's Center for Strategic Leadership: The Information Paradox: Realizing the Benefits of Information Tech., McGraw-Hill, 1998
- [31] US-CERT Vulnerability Database, US Computer Emergency Readiness Team, <http://www.kb.cert.org/vuls/> (current 11/2006)
- [32] US. General Accounting Office, Information Security Risk Assessment: Practices of Leading Organizations, 1999
- [33] D. Verdon, G McGraw, Risk analysis in software design, Security & Privacy Magazine, IEEE Volume 2, Issue 4, Jul-Aug 2004 Page(s):79 - 84
- [34] Y. Yang, J. Bhuta, D. N. Port, B. Boehm, Value-Based Processes for COTS-Based Applications, IEEE Software, July-August 2005