Probabilistic Validation of Intrusion Tolerance[1]

William H. Sanders*, Michel Cukier**, Franklin Webber***, Partha Pal***, and Ronald Watro***

*Coordinated Science Laboratory and
Electrical and Computer Engineering Department
University of Illinois at Urbana-Champaign
whs@crhc.uiuc.edu

**Center for Reliability Engineering
Department of Materials and Nuclear Engineering
University of Maryland at College Park
mcukier@eng.umd.edu

***BBN Technologies
{fwebber, ppal, rwatro}@bbn.com

1. Introduction

Intrusion tolerance is an emerging approach to security that aims to increase the likelihood that an application will be able to continue to operate correctly in spite of malicious attacks that may result in successful intrusions. Most traditional approaches to security validation have not been quantitative, instead focusing on specifying procedures that should be followed during the design of a system (e.g., the Security Evaluation Criteria [DOD85, ISO99]). When quantitative methods have been used, they have typically either been based on formal methods (e.g., [Lan81]), aiming to prove that certain security properties hold given a specified set of assumptions, or been quite informal, using a team of experts (often called a "red team," e.g. [Low01]) to try to compromise a system. An alternative approach has been to try to quantify, probabilistically, the behavior of an attacker and his impact on the ability of a system to provide certain security-related properties.

In this extended abstract, we first (in Section 2) review existing probabilistic approaches. We then (in Section 3) describe work we are doing in this area, with the goal of creating a sound scientific basis for comparing alternative intrusion tolerance approaches quantitatively, and estimating the intrusion tolerance of particular approaches. Our main measure of security is application-level availability, which we define as a measure of correct delivery of service with respect to the alternation of correct and incorrect service [Lap91]. Realizing this goal will require work both in modeling and measurement, and the creation of guidelines for their application in intrusion tolerance approaches.

2. Existing Probabilistic Approaches

Early work on probabilistic quantification of security was done by Littlewood et al. [Lit93]. The goal of this work was to investigate the similarities between dependability and security in order to define measures of "operational security" similar to those used in dependability evaluation. In doing so, Littlewood et al. made the important observation that effort was an appropriate index for expressing security measures.

Jonsson et al. [Jon97] presented a quantitative model of the security intrusion process based on attacker behavior. Jonsson conducted several experiments to build a model of typical attacker behavior, postulating that a process representing the behavior of an attacker can be split into three phases: the learning phase, the standard attack phase, and the innovative attack phase.

Several attempts have also been made to build models that take into account both the attacker and the system being validated. Specifically, Gong et al. [Gon01] present a general 9-state model of an intrusion-tolerant system for describing known and unknown security exploits by considering the impacts of attacks, rather than explicitly representing vulnerabilities that can lead to intrusions. The attacker and system are not represented explicitly in the model; instead, the model represents the state of the system in terms of (high-level) events that lead to failures.

[Jha01] proposes that state-level modeling, formal logic, and a Bayesian analysis be used together to quantify the survivability of a system. The authors first model the network nodes and links of a networked system using state machines. Faults are then injected in the models; links are assumed to be non-faulty, and the nodes are assumed to be non-faulty, faulty, or intruded. The third step consists of specifying a survivability property, e.g., the system enters a faulty state, using a temporal logic. From the state machine model with the injected faults and the survivability property, the authors then generate a scenario graph. The scenario graph is then used for evaluating the overall system reliability or the latency using Bayesian networks.

Finally, Ortalo et al. [Ort99] propose modeling known vulnerabilities in a system using a "privilege graph" (a privilege graph is similar to the scenario graph described above). By combining a privilege graph with simple assumptions concerning an attacker's behavior, the authors then obtain an "attack state graph." Parameter values for attack state graphs are presumed to have been obtained experimentally; once obtained, an attack state graph can be analyzed using standard Markov techniques to obtain several probabilistic measures of security. To illustrate the use of their approach, the authors present an analysis using an attack state graph built using data obtained from measurements taken on a large computer installation over a 21-month period.

3. Outline of Proposed Approach

Organizationally, we believe that a probabilistic validation of security with respect to availability should have two components: 1) a *model*, of an attacker, the system, and the workload demanded of a system, and 2) a set of *measure-*

[1] This research has been supported by DARPA Contract F30602-00-C-0172.

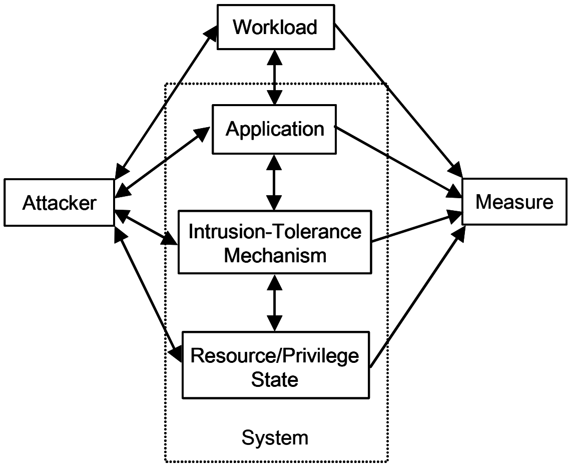

Figure 1: Probabilistic Security Model Structure

ments that provide estimates of the values of model parameters.

More specifically, for intrusion-tolerant systems, we envision the creation of a model with the structure depicted in Figure 1. In particular, we believe that a probabilistic model of an intrusion-tolerant system can be broken down into the following submodels: an *Attacker submodel*, a *Workload submodel*, an *Application submodel*, an *Intrusion-Tolerance Mechanism submodel*, and a *Resource/Privilege State submodel*. The arcs connecting submodels in the figure represent possible interactions between submodels that can change their state. For example, the attacker may be able to change the state of a resource or amount of privilege granted to him (as represented by the directed arc from the attacker to the Resource/Privilege State submodel), or the attacker may change his state (and hence change his behavior) by using knowledge he has gained by observing the state of the system (represented by the directed arcs from the system submodels to the attacker.)

In each model, determining the appropriate level of detail/abstraction is very important, and depends on the scope and purpose of the model. For example, the system submodels should represent the parts of a system that are important, relative to the types of attacks considered and the expression of a particular availability measure. In particular, they must be detailed enough to support the expression of those parts of state that an attacker may change and those parts that may change his behavior. Depending on the nature of the attack, the attacker model may either represent details of the intrusion itself or represent the *effect* of the intrusion.

Likewise, the system submodels must be detailed enough to support the expression of the availability measure considered. For example, Ortalo et al. [Ort99] suggest that an appropriate notion of state for a probabilistic system model would be the *degree of privilege* that an attacker has attained. In addition to representing the degree of privilege, we believe that it is also important to represent resources in the system that are necessary for the application to function, since our ultimate goal is to quantify the availability as perceived by a user of an application. The two state aspects will be combined in the Resource/Privilege State submodel.

The level of detail/abstraction also depends on the input parameter values (obtained from measurement data) available for each model. The type and accuracy of input parameter values will depend on the stage of development of the system that is being validated. For existing systems, we could use methods similar to Jonsson's to obtain values that quantify the behavior of an attacker. Those values could then be used to build a model of the attacker. Concerning the Resource/Privilege State model, we intend to use an up-to-date network security scanner like Nessus [Nessus] combined with a security auditing tool like COPS [COPS] to obtain the parameter values. The application, intrusion tolerance mechanism, and workload models are more case-specific, and require further study to determine appropriate input parameters and their values. We therefore intend to explore the relevant level of detail for the model, input values, and measures by focusing on several case studies.

The goal of the presented validation framework combining models and measurements is to create a sound scientific basis for comparing alternative intrusion tolerance approaches quantitatively, and estimating the intrusion tolerance of particular approaches.

References

[COPS] http://www.fish.com/cops/

[DOD85] U.S. Department of Defense Standard, "Department of Defense Trusted Computer System Evaluation Criteria" ("Orange Book"), DOD 5200.28-STD, Library No. S225,7ll, Dec. 1985. http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html

[Gon01] F. Gong, K. Goseva-Popstojanova, F. Wang, R. Wang, K. Vaidyanathan, K. Trivedi, and B. Muthusamy, "Characterizing Intrusion Tolerant Systems Using A State Transition Model," in Proc. DARPA Information Survivability Conference and Exposition II. DISCEX'01, 2001.

[ISO99] ISO/IEC International Standards (IS) 15408-1:1999, 15408-2:1999, and 15408-3:1999, "Common Criteria for Information Technology Security Evaluation": Part 1: "Introduction and General Model," Part 2: "Security Functional Requirements," and Part 3: "Security Assurance Requirements," Version 2.1, August 1999 (CCIMB-99-031, CCIMB-99-032, and CCIMB-99-033). http://csrc.nist.gov/ cc/ccv20/ccv2list.htm

[Jha01] S. Jha and J. M. Wing, "Survivability Analysis of Networked Systems," in *Proc. of the 23rd International Conference on Software Engineering (ICSE 2001)*, p. 307-317, 2001.

[Jon97] E. Jonsson and T. Olovsson, "A Quantitative Model of the Security Intrusion Process Based on Attacker Behavior," *IEEE Transactions on Software Engineering*, vol. 23, no. 4, pp. 235-245, April 1997.

[Lan81] C. Landwehr, "Formal Models for Computer Security," *Computer Surveys*, vol. 13, no. 3, Sept. 1981.

[Lap91] J. C. Laprie, ed., *Dependability: Basic Concepts and Terminology*, Vol. 5 of *Dependable Computing and Fault-Tolerant Systems*, Springer-Verlag, 1991.

[Lit93] B. Littlewood, S. Brocklehurst, N. Fenton, P. Mellor, S. Page, D. Wright, J. Dobson, J. McDermid, and D. Gollmann, "Towards Operational Measures of Computer Security," *Journal of Computer Security*, vol. 2, no. 2-3, pp. 211-229, 1993.

[Low01] J. Lowry, "An Initial Foray into Understanding Adversary Planning and Courses of Action," in *Proc. DARPA Information Survivability Conference and Exposition II (DISCEX'01)*, pp. 123-133, 2001.

[Nessus] http://www.nessus.org/

[Ort99] R. Ortalo, Y. Deswarte, and M. Kaaniche, "Experimenting with Quantitative Evaluation Tools for Monitoring Operational Security," *IEEE Transactions on Software Engineering*, vol. 25, no. 5, pp. 633-650, Sept.-Oct. 1999.

[Rit00] R. W. Ritchey and P. Ammann, "Using Model Checking to Analyze Network Vulnerabilities," in *Proc. 2000 IEEE Symposium on Security and Privacy (S&P 2000)*, pp. 156-165, 2000.