

Modeling SITAR System Security

Dazhi Wang¹, Bharat B. Madan², and Kishor S. Trivedi²

¹Department of Computer Science, Duke University
wangdz@cs.duke.edu

²Department of Electrical and Computer Engineering, Duke University
{bbm, kst}@ee.duke.edu

I. INTRODUCTION

Recent strategies to protect system security lay emphasis on designing intrusion-tolerant systems that are able to tolerate intrusions using techniques such as redundancy, diversity, reconfiguration and graceful degradation. These systems are expected to not only detect and tolerate attacks, but also repair, or rejuvenate themselves so as to remove any damage caused by an intrusion. Several research efforts are currently afoot to design such systems, such as Enclaves, EMERALDS, ITUA, MAFTIA and SITAR.

Before any security mechanism can be accepted to provide protection to a system, it is important to assess its efficacy. Earlier security evaluations were mainly based on a qualitative assessment, such as [1], [2]. This may not be enough to characterize intrusion tolerant systems. Recent studies to security evaluation have begun to take a quantitative approach by using probabilistic and statistical methods as in traditional reliability analysis [6], [3], [5], [4]. In this paper we apply probabilistic modeling to the SITAR system, which is an intrusion tolerant architecture developed jointly by MCNC and Duke University. We start with a continuous-time Markov model that describes the dynamic behavior of multiple intrusion tolerance strategies that exist in SITAR. In order to increase the fidelity of the model to the SITAR architecture, we found it difficult to continue with the hand construction of a CTMC, we motivate and use the stochastic reward net (SRN) model to capture the SITAR system behavior as well as the attacker behavior.

II. SITAR SYSTEM OVERVIEW

The SITAR system has the following main subsystems: proxy servers, acceptance monitors, ballot monitors, audit control module, adaptive reconfiguration module (ARM), and the COTS servers. Figure 1 presents a logical view of the SITAR architecture. In SITAR, the detection of an attack and the compromise can be carried out by multiple subsystems as enumerated below:

- 1) Malicious request detection by acceptance monitor

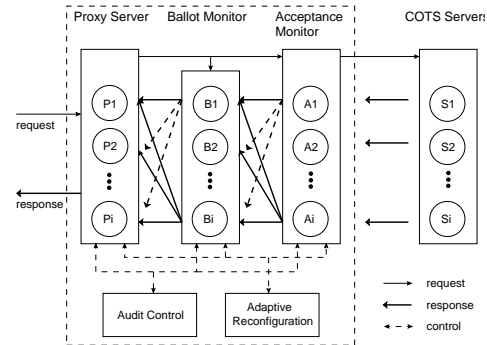


Fig. 1. SITAR System Architecture

- 2) Acceptance testing on the responses by acceptance monitor
- 3) Voting and agreement by ballot monitor
- 4) Automatic auditing in audit control module and manual monitoring

III. THE BASIC MODEL

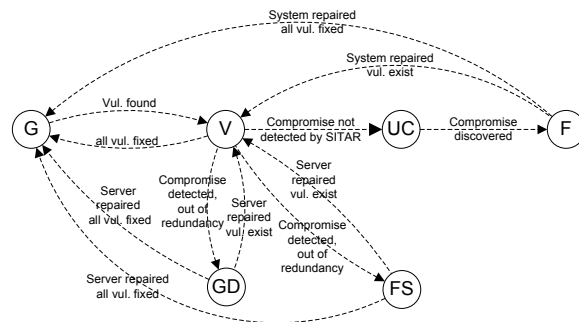


Fig. 2. State Transition Model for Threat Level One

Figure 2 and Figure 3 shows the state transition diagram for the SITAR system operating in an environment with varying threat levels. In threat level 1, there is only 1 COTS server while for the threat level 3 there are 3 redundant COTS servers so as to achieve higher security

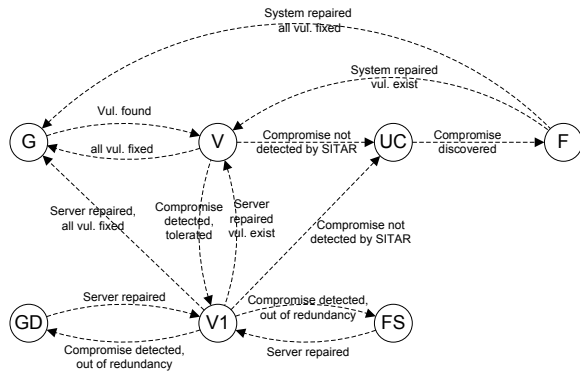


Fig. 3. State Transition Model for Threat Level Three

and intrusion tolerance capabilities. Initially, the system is in state G , in which no vulnerability is found. The system will go into vulnerable state V from good state G , indicating that some of these vulnerabilities have been identified. In state V , attackers try various ways to exploit these vulnerabilities, and in the mean time system administrators may be fixing the vulnerabilities. If all vulnerabilities are fixed before the attacker can issue an attack, the system will return to state G ; otherwise the vulnerability will eventually be successfully exploited and an attack is injected into the system.

If SITAR's internal components are able to detect an attack before any damage gets done to the system, the system will continue to stay in vulnerable state V . Otherwise the attack sneaks into the system resulting in some damage to the COTS servers. If the compromise is detected, and the resulting damage can be tolerated by redundancies present in the SITAR system, the system still returns to the vulnerable state after the attack. As is the case in Figure 3, if one COTS server is compromised by the malicious attack and the system detects the compromise, it can tolerate the compromise by changing to state V_1 and continue providing services with 2 COTS servers up. If the system detects the compromise but can not mask the damage the attack causes, it will enter either graceful degradation state GD or fail-secure state FS depending on the impact of the attack.

IV. SRN MODEL

In order to capture the details of attacker behavior as well as system behavior, and to avoid the manual construction of a high fidelity but complex Markov chain, we develop the SRN model for SITAR security. Figure 4 gives the model. Places P_G , P_V , P_{GD} , P_{FS} , P_{UC} and P_F correspond to the system's security states. Transitions T_{mal} and T_{att} are timed transitions characterizing the attackers' behavior of probing the system's vulnerabilities and their subsequent exploitation to attack the system. Places P_A , P_{am} , P_{bm} , P_{up} and P_{down} and the transitions

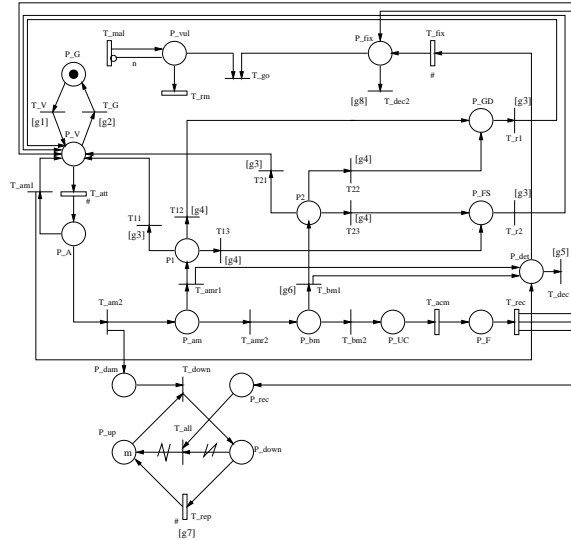


Fig. 4. SRN Model for SITAR Security

associated with these places characterize the system's redundancy structure and intrusion tolerance mechanisms. Finally the transitions T_{rep} , T_{rec} , T_{fix} characterize the system recovery behavior. The guard functions are shown in Table I.

	guard functions
g1	$\#(P_{vul} \neq 0)$
g2	$\#(P_{vul} = 0)$
g3	$\#(P_{up}) > \#(P_{down})$
g4	$\#(P_{up}) < \#(P_{down})$
g5	$\#(P_{det}) > \#(P_{vul})$
g6	$\#(P_{up}) > 1$

TABLE I

ENABLING FUNCTIONS IN SITAR SRN MODEL

REFERENCES

- [1] *Information Technology Security Evaluation Criteria (ITSEC):Provisional Harmonized Criteria*. ISBN 92-826-7024-4, Dec. 1993.
- [2] P.D. Goldis. Questions and answers about tiger teams. *EDPACS, The EDP Audit, Control and Security Newsletter*, 27(4):1–10, Oct. 1989.
- [3] E. Jonsson and T. Olovsson. A quantitative model of the security intrusion process based on attacker behavior. *IEEE Transactions on Software Engineering*, 23(4):235–245, April 1997.
- [4] B.B. Madan, K. Goševa-Popstojanova, K. Vaidyanathan, and K.S. Trivedi. Modeling and quantification of security attributes of software systems. In *Proc. Int. Conf. on Dependable Systems and Networks, (IPDS stream)*, volume 2, pages 505–514, 2002.
- [5] R. Ortalo, Y. Deswarte, and M. Kačičič. Experimenting with quantitative evaluation tools for monitoring operational security. *IEEE Transactions on Software Engineering*, 25:633–650, Oct. 1999.
- [6] S. Singh, M. Cukier, and W. H. Sanders. Probabilistic validation of an intrusion-tolerant replication system. *International Conference on Dependable Systems and Networks (DSN'03)*, June, 2003.