DNS Spoofing Attack Simulation for Model-Based Security Evaluation

Golriz Khazan[1] and Mohammad Abdollahi Azgomi[2]

[1]ICT Group, E-Learning Center, Iran University of Science and Technology, Tehran, Iran
E-mail: khazan.golriz@gmail.com
[2]Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
E-mail: azgomi@iust.ac.ir

Abstract. Security of computer systems and networks has become very significant nowadays. Introducing and using a unified framework for modeling and quantitative security evaluation (QSE) is an open problem. Th results of our study on drawbacks of the existing security assessment methodologies motivated us to use a simulation framework for model-based security evaluation. We have used discrete-event simulation (DES) and the *SimEvents* tool for QSE of a domain name system (DNS). First, the normal operation of the DNS is simulated. Then, an attacker is added to the model. The aim is to evaluate the instantaneous availability of DNS as an important measure of security. Finally, as a case study, DNS spoofing attack model is constructed and the availability of the attacked system is evaluated. The proposed approach can be used for other kinds of attacks and other types of systems, networks and applications. In this paper the simulation models and their results are presented.

Keywords: Discrete-event simulation (DES), quantitative security evaluation (QSE), model-based security evaluation, SimEvents.

1. Introduction

A computer security incident is a change of state in a bounded computer system from the desired state to an undesired state, where the state change is caused by the application of a stimulus external to the system [1]. This state change is issued by an external irritant application to the system. Security brings in concerns for confidentiality, in addition to availability and integrity. Basic definitions are given first and then commented upon. Next they are supplemented by additional definitions, which address the threats to dependability and security (faults, errors, failures), their attributes, and the means for their achievement (fault prevention, fault tolerance, fault removal, fault forecasting).

Several methods and checklists, such as information technology security evaluation criteria (ITSEC) [2] and common criteria (CC) [3], etc. are widely used for assessing the security of computer systems and networks. However, some of the disadvantages of these methods include the cost of their usage and the time consumed for their accomplishment. Another important disadvantage is that these methods are not applicable in design and development phases of systems.

There are two types of dependability evaluation methods:

- O*rdinal evaluation*, which identifies, classifies, and ranks the failure modes or the event combinations component failures or environmental conditions) that will lead to system failures, like *failure modes, effects and critically analysis* (FMECA), *reliability block diagrams* (RBD) and *fault trees*.
- *Probabilistic evaluation* that evaluates in terms of probabilities the extent to which some of the attributes (measures) are satisfied, like Markov chains, stochastic Petri nets (with analytical solutions or simulation).

The efforts on *quantitative security evaluation* (QSE) of systems are typically based on Markov chains, formal methods and red-teams, but introducing and using a unified framework for modeling, simulation and QSE is an open problem.

SimEvents [4] extends *Simulink* with a discrete-event simulation model of computation. With *SimEvents*, activity-based models of systems are developed to evaluate system parameters, such as congestion, resource contention and processing delays. It is possible to configure entities with user-defined attributes and then aggregate entities and attributes to model data hierarchy and transport in applications such as packet-based networks, mission planning, supervisory control, real-time operating systems and computer architecture.

In this paper, simulation of a domain name system (DNS) for QSE based on discrete-event simulation (DES) using *SimEvents* is presented. First, the system is in normal state and then the arrival of an attacker is simulated and the availability of the system, as an important security measure, in any moment of simulation time is evaluated. Finally, a case study of DNS spoofing attack simulation is done and the availability measure of the system is evaluated.

The rest of this paper is organized as follows. In section 2, some related works are reviewed. In section 3, a simulation model for evaluation of DNS system availability is presented. In section 4, as a case study, DNS spoofing attack simulation model is presented. First, the system operated normally and then the arrival of attacker is modeled. Finally, some concluding remarks are mentioned in section 5.

2. Related Works

The use of simulation for QSE can solve the problems and drawbacks of the existing methods. Till now, the common application of simulation in security is to use of simulation tools in ordinary networks to model systems and traffics that demonstrates attacks. By security measurements affecting performance of systems, a useful and obvious way for simulation is acquired; however, the suggested security measurements are limited and expensive for the specified applications.

As the related works, we can mention examples of Nicol and his colleagues who have worked directly on border gateway protocol (BGP) including the Internet. The BGP example consists of encryption and decryption which takes time and can thus affect performance. Additionally the BGP system is so large that one should use simulation to capture the system dynamics [8, 9]. In [12] as a first step towards security quantification the similarities between reliability and security from the perspective of evaluating measures of operational security of systems is discussed. In [13] a quantitative model to measure known UNIX security vulnerabilities using a privilege graph is represented, which is transformed into a Markov chain. Gupta and his colleagues tried to evaluate security and performance of several intrusion-tolerant architectures in [5]. Other related works in this field are the probability security meter model introduced by Sahinoglu, that receives inputs like vulnerability, threat, lack of countermeasure and constants like criticality, utility cost and then measures residual risk and cost needs for avoiding from risk [10]. Simulation-based analysis of security is used in mobile ad-hoc networks (MANET) is another activity in this field in which the impact of network performance threats by means of dynamic resource routing (DSR) is studied [11]. A survey over the existing model-based system dependability evaluation techniques is provided in [6], and summarizes how they are being extended to evaluate security. In [14, 15] the use of stochastic modeling technique is suggested as a suitable method for assessing the trustworthiness of a system, regardless of whether the failure cause is intentional or not. Security consideration as a quality of service attribute an d an approach to quantify security attributes of intrusion tolerant systems using stochastic modeling techniques is presented in [16, 17]. Finally, in [1], coloured Petri nets (CPNs) and the CPN Tools are used for modeling and quantitative risk analysis of enterprise systems.

3. A Simulation Model for Evaluation of System Availability

In this section, a simulation model is presented, which is constructed by SimEvents tool. The aim is to evaluate the instantaneous availability as an important security measure of a domain name system (DNS). As shown in Fig. 1, the system is initially in its normal state without being under any attack. Then, a subsystem as *security failure* is injected. Security failures are supposed to be intentional. Therefore, the availability of the system, as a quantitative security parameter, can be measured.

In our model, the *Entity Time-Based Generator* block generates failure entities with specified exponential distribution, mean=8000 (a system with exponential distribution mean=8000 fails). When the system detects the security failures, it starts to fix them. The time that the system consumes to repair the security failures can be varied, therefore in the model, a *Random Number Event Generator* with uniform distribution between 10 and 70 is used to generate random number as a repair time. This number is used as the *service time to failure* entity or the *system failure time*. After failure entity arrival, server begins to repair the failure and will not serve other entities. When the lower queuing system generates an entity, changes in its server's #*n* signal invoke the state flow block that determines the state of the upper queuing system. Increases in the #*n* signal causes the server to go down, while decreases causes the state flow output signal, *Server-Up* becomes one and registers in memory by a latch and then gate enables. Simultaneously, *Server-Down* becomes zero and registers in another *Signal Latch.* So, the output of upper *Signal Latch* enables the gates named *Enabled Gate1* and *Enabled Gate* then the entities (or packets) from sources waiting in their queues arrive to DNS System. DNS System is simulated by

Path Combiner and *Output Switch* blocks. Then the entities are sent to right destination. Output of this model is the system availability ($A(t)$) of true destination and attacker system. By increasing the mean value of failure entity generator, $A(t)$ of true destination will decrease and entities will send to attacker so the A(t) of attacker system will increase. The result is shown in Fig. 2 and Fig. 3.

The goal of many attacks is to sabotage and spoof servers in network systems. In this study we have focused on DNS spoofing. In DNS spoofing domain address sent by the user to DNS server is intercepted by the attacker and is translated into a fake IP address and then will be sent back to the user. Based on this scenario, the user trusts the attacker and creates a TCP connection with the attacker. Finally, the attacker discloses confidential data of the user by exploiting of this connection [7, 18].

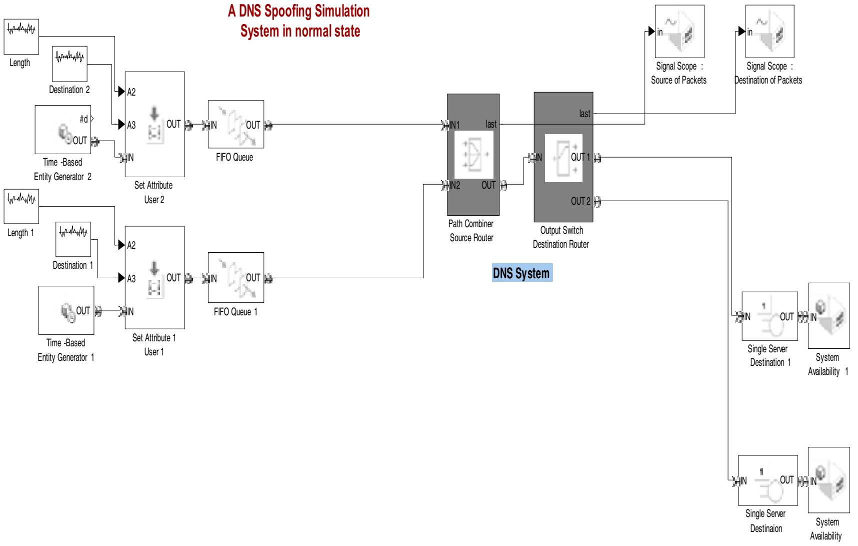

Fig. 1. Simulation model for evaluating availability of true destination and attacker server

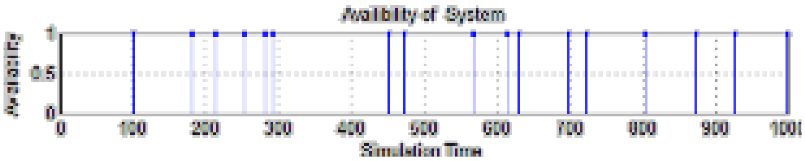

Fig. 2. Availability of system in normal state with probability of Security failures

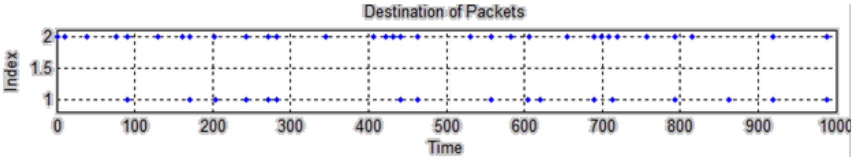

Fig. 3. Packets destination in normal state

4. Case Study: Modeling and Evaluation of DNS Spoofing

In this section, as a case study, an intrusion process is modeled and evaluated by a simulation model. The descriptions and scenario of the attack is same as [18].

4.1 Simulation Model of the Normal Operation

The model shown in Fig. 4 consists of two clients in the left, a path combiner and an output switch as a DNS in the middle, and two servers as destination in the right. These clients generate packets with specified length and send them to a specified target server nominated in its packet. Packets are generated by *Time-Based Entity Generator* with mean=50 and 200 from SimEvents library. Packets length and destinations are generated by *Random Number Event-Based Generator* with uniform distribution. Destination and length can be a number between 1 and 2, 6 and 10, respectively. Destinations are target servers. To set the specification of each client, *Set Attribute* blocks are used. First attribute, *A1* or *Source* is source packet generator for 1st client and is set to one and for the 2nd one, it is set to two. Second attribute, *A2* or *Length* is packet length that its value is specified from random number generator block connected to set attribute block. Packets after generating are routed by source router to the destination server that is specified in set attribute block of its own generator. When packets were generated, they were stored in their limited-capacity queue in order to hold and guide packets. Capacity of both queues is 25 packets and no preemption is defined for passing packets from queues.

To simulate DNS, we use path combiner and output switch, which receives packet from input ports, find the right destination and leads them to target servers based on their destination. For each packet, destination can be one of the 1 or 2 servers. Service time of servers is adjusted to 10. After processing packets in server, they have been led to *Entity Sink* or used to measure and report system parameters. First, the server output is used for measuring *A*(*t*) of the first server. These simulation outputs are four axes. One spots generated packet from two clients, the other one shows received packets and the two lasts demonstrates availability of servers. As shown in Fig. 4, the first server is busy in 100, 180… 210… 1000 simulation times, so it is idle in 10, 20… 70… 990. The second server operates similarly.

4.2 Simulation Model of the DNS Spoofing Attack

To illustrating spoofing in our model, we suppose that the second destination server is an attacker who wants to spoof the DNS system in order to lead packets of the first source to him/her. To simulate security failure and consequently a chance for spoofing DNS, we injected a security failure subsystem to our model. The security failure subsystem consists of (i) a *Time-based Entity Generator* that generates security failure entities with exponential distribution mean=1000 (in attack time), (ii) a security failure repair server which receives the repair time from a *Random Number Generator* with uniform distribution between 10 and 70.

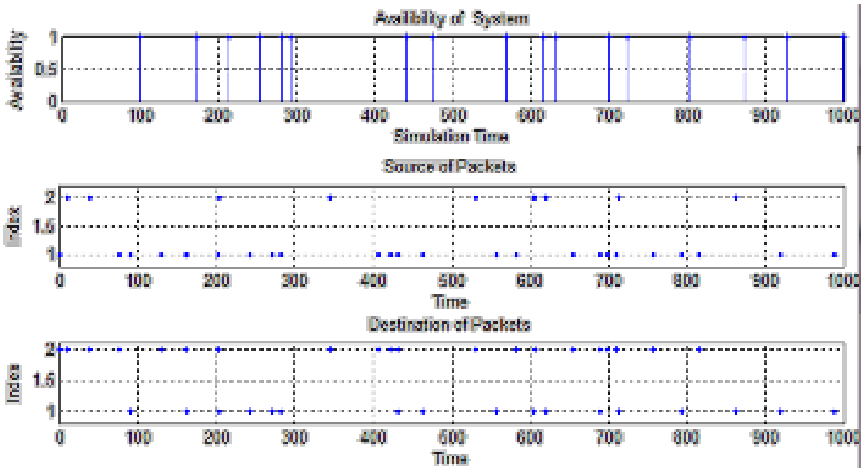

Fig. 4. Simulation results for 1000 run-times

After failure entity arrival, server begins to repair the failure and will not serve other entities. When the lower queuing system generates an entity, changes in its server's #*n* signal invoke the state flow block that determines the state of the upper queuing system. Increases in the #*n* signal causes the server to go down, while decreases causes the state flow output signal, *Server-Up* becomes one and registers in memory by a latch and then gate enables. Simultaneously, *Server-Down* becomes zero and registers in another *Signal Latch.* So, the output of upper *Signal Latch* enables the gates named *Enabled Gate1* and *Enabled Gate* then the entities (or packets) from sources waiting in their queues arrive to DNS System. When the server goes down it means that DNS is spoofed so by using a *Set Attribute* block the destination packet value of first client (Attacker Target) will be changed to which attacker wants meaning second destination server instead of first destination server and then modified packets via a P*ath Combiner* are led to DNS.

DNS System is simulated by *Path Combiner* and *Output Switch* blocks. Path Combiner receives packets from input port and then sends them to destination by using *Output switch*. In the meantime, DNS is maybe spoofed and led packets to attacker.

As shown in Fig. 5, this simulation model exposes a *DNS Spoofing.* First, in normal state, generating packets mean value of first source (Attacker Target) is 50, and then by decreasing the mean of generating security failure entity generator, attack happens and packets from first server are sent to attacker server so the availability of attacker server in all of the simulation time becomes one. Moreover, other outputs of this simulation illustrating the packets departed from the sources and destinations are shown.

After 1000 run times of simulation and setting the security failure mean value to 8000, the output will be as Fig. 6. Now, we decrease the mean value to 1000 and run simulation for 1000 times again. Considering that during the total simulation time, the attacker took the control of victim server and packets are sent to attacker server instead of proper target. The results are shown in Fig. 6.

5. Conclusions

In this paper, discrete-event system simulation and *SimEvents* is used for quantitative security evaluation. The proposed model, measures the availability of DNS as one of the quantitative security measures, before and after attack. The simulation results show that by increasing the rate of generating security failure entities, the availability of correct server decreases so attacker server receives packets henceforward.

As a case study, we constructed a simulation model for DNS spoofing attack. First, clients send packets to servers and the attacker as a client intrudes to the system and spoofs victim DNS and receives packets. During the simulation, the availability measure, $A(t)$, of the server is measured.

As a future work, we can model and evaluate the availability measure in large and sophisticated computer and communication systems to see the potential benefits of the proposed simulation methodology.

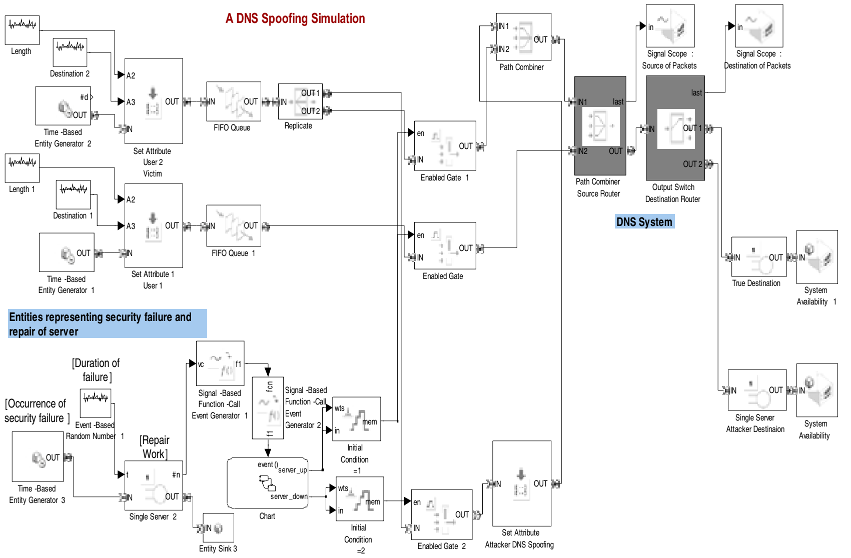
A DNS Spoofing Simulation
Length
Destination 2
Time -Based Entity Generator 2
Set Attribute User 2 Victim
FIFO Queue
Replicate
Length 1
Destination 1
Time -Based Entity Generator 1
Set Attribute 1 User 1
FIFO Queue 1
Path Combiner
Enabled Gate 1
Enabled Gate
Path Combiner Source Router
Output Switch Destination Router
Signal Scope : Source of Packets
Signal Scope : Destination of Packets
DNS System
True Destination
System Availability 1
Single Server Attacker Destinaion
System Availability
Entities representing security failure and repair of server
[Duration of failure]
[Occurrence of security failure]
Event -Based Random Number 1
Time -Based Entity Generator 3
[Repair Work]
Single Server 2
Entity Sink 3
Signal -Based Function -Call Event Generator 1
Signal -Based Function -Call Event Generator 2
Chart
server_up
server_down
Initial Condition =1
Initial Condition =2
Enabled Gate 2
Set Attribute Attacker DNS Spoofing

Fig. 5. DNS spoofing attack simulation model

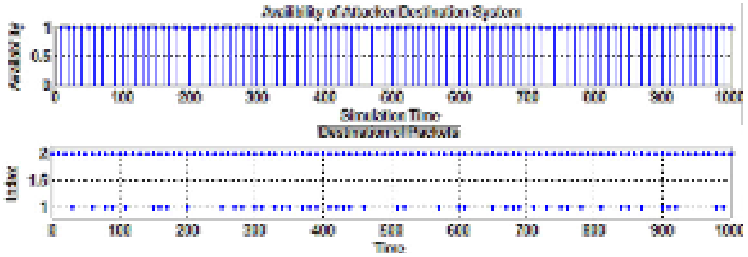

Fig. 6. Simulation results for 1000 run-times and security failure mean=1000

References

1. P. R. Stephenson, "A Formal Model for Information Risk Analysis Using Colored Petri Nets," Proc. of the 5th Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, University of Aarhus, Denmark, October 8-11 (2004)
2. "Information Technology Security Evaluation Criteria", URL: `http://www.bsi.de/zertifiz/itkrit/itsec-en.pdf`
3. "Common Criteria", URL: `http://www.commoncriteriaportal.org`
4. "SimEvents," URL: `http://www.mathworks.com`
5. V. Gupta, V. V. Lam, H. V. Ramasamy, W. H. Sanders, and S. Singh, "Dependability and Performance Evaluation of Intrusion-Tolerant Server Architectures," Proc. of the 1st Latin-American Symp. on Dependable Computing (LADC'03), LNCS 2847, Springer (2003) 81-101
6. D. M. Nicol, W. H. Sanders, and Trivedi K. S., "Model-Based Evaluation: From Dependability to Security," IEEE Transactions on Dependable and Secure Computing, Vol. 1, (Jan.-March 2004) 48-65
7. E. Malekian, Network Intrusion and Countermeasures, Nas Publications (2004) (*in Persian*)
8. D. M. Nicol, S. W. Smith, M. Zhao, "Evaluation of Efficient Security for BGP Route Announcements using Parallel Simulation", *Simulation Modeling Practice and Theory*, Vol. 12 (2004) 187-216
9. D. Nicol, S.W. Smith, "Modeling and Simulation in Security Evaluation", *IEEE Security & Privacy* (2005)
10. Sahinoglu, M., Trustworthy Computing: Analytical and Quantitative Engineering Evaluation, Wiley-Interscience (2007)
11. S. Porcarelli, F.D. Giandomenico, A. Bondavalli, and P. Lollini. "Model-Based Evaluation of a Radio Resource Management for Wireless Networks," *Proc. of the Computing Frontiers*, Ischia, Italy (April 2004)
12. B. Littlewood, et al. "Towards Operational Measures of Computer Security," *Journal of Computer Security*, Vol. 2 (Oct 1993) 211-229.
13. R. Ortalo, et al., "Experiments with Quantitative Evaluation Tools for Monitoring Operational Security*", IEEE Trans. Soft. Eng.,* Vol. 25, No. 5 (Sept./Oct. 1999)
14. K. Sallhammar, et al., "On Stochastic Modeling for Integrated Security and Dependability Evaluation" *Journal of Networks*, Vol. 1, No.5 (Sept./Oct. 2006)
15. K. Sallhammar and S. J. Knapskog , "Using game theory in stochastic models for quantifying security", *Proc. of the 9th Nordic Workshop on Secure IT-Systems*, Espoo, Finland, November 4-5 (2004)
16. D. Wang, et.al. "Security Analysis of SITAR Intrusion Tolerant System," *Proc. of ACM Workshop on Survivable and Self-Regenerative Systems* (2003) 23-32
17. K. B. Madan, et.al. "A Method for Modeling and Quantifying the Security Attributes of Intrusion Tolerant Systems," *Performance Evaluation,* Vol. 56 (2004)
18. J. Almasizadeh and M. Abdollahi Azgomi, "A New Method for Modeling and Evaluation of the Probability of Attacker Success," *Proc. of the 2008 International Conference on Security Technology (SecTech'08)*, Horizon Resort, Sanya, Hainan Island, China, Dec. 13-15, IEEE CS Press (2008) *to appear*