Network Security Validation
Using Game Theory

Vicky Papadopoulou and Andreas Gregoriades
Computer Science and Engineering Dep., European University Cyprus, Cyprus
{v.papadopoulou,a.gregoriades}@euc.ac.cy

Abstract: Non-functional requirements (NFR) such as network security recently gained widespread attention in distributed information systems. Despite their importance however, there is no systematic approach to validate these requirements given the complexity and uncertainty characterizing modern networks. Traditionally, network security requirements specification has been the results of a reactive process. This however, limited the immunity property of the distributed systems that depended on these networks. Security requirements specification need a proactive approach. Networks' infrastructure is constantly under attack by hackers and malicious software that aim to break into computers. To combat these threats, network designers need sophisticated security validation techniques that will guarantee the minimum level of security for their future networks. This paper presents a game-theoretic approach to security requirements validation. An introduction to game theory is presented along with an example that demonstrates the application of the approach.

1. Introduction

In recent years organizations have experienced the explosion of attacks on their information resources. Among all these attacks, computer virus poses a major threat to the information security for business effectiveness. According to the Computer Security Institute (CSI) [8], viruses constitute the principal cause of financial losses among computer security incidents in organizations. Given this, there is a major need to understand and control virus behaviour in network-centric information systems. A computer network is defined as the purposeful interconnection of computer nodes for the efficient and effective interchange of information. Network security consists of the provisions enforced in a computer network that aim to protect the network and its resources from unauthorized access. The recent growth of public networks such as the Internet made this requirement even more critical. However, the dynamic characteristics of contemporary networks combined with their increased size create an extra challenge for the network designers. This area of research has gained considerable popularity due to the implications it has on users' satisfaction and business reputation. Therefore, being able to quantify the security performance of a future network early in the design phase is of vital importance. The need to validate security requirements early has been addressed also by Lamsweerde [3] and Crook [1]. Security as an NFR is influenced by functional aspects of the system. Unlike functional requirements, which can be deterministically validated, NFRs are soft variables that cannot be implemented directly; instead, they are satisfied [1] by a combination of functional requirements. NFRs define the overall qualities or attributes of the resulting system and as such place restrictions on the software product being developed. Typical approaches to validating NFRs include formal methods, prototypes and system simulations [2] and use of scenarios. Scenarios describe all the states that the network could have, given all the combinations of attack behaviours of the viruses. Specifically, the application of scenarios-based approaches highlighted the problem of having too many scenarios to analyse. This paper addresses this problem through Game Theory. Specifically, we reduce the complexity of the solution space to a manageable set and hence escape from the problem of evaluating too many scenarios.

2. Game Theory

Game Theory attempts to mathematically model the rational behaviour of actors in strategic situations. In such situations actor's success depends on the choices of others. Most of the existing and foreseen complex networks, such as the Internet, are operated and built by thousands of large and small entities (autonomous *agents*), which collaborate to process and deliver end-to-end flows originating from and terminating at any of them. The distributed nature of the Internet implies a lack of coordination among its users that attempt to maximise their performance according to their own parameters and objectives. Recently, Game Theory has been proven to be a powerful modelling tool to describe such *selfish*, rational and at the same time, decentralized interactions. Game Theory models such interactions as players with potentially different goals (*utility functions*), that participate under a common setting with well prescribed interactions (*strategies*), e.g. TCP/IP protocols. The core concept of Game Theory is the notion of *equilibrium* that is defined as the condition of a system in which competing influences are

balanced. A *game*, expressed in normal form is given by a tuble $G=(M, A, \{u_i\})$, where G is a particular game, M is a finite set of players (decision makers) $\{1,2,…,m\}$, A_i is the set of actions available to player i, $A = A_1 \times A_2 \times \cdots \times A_m$ is the action space, and $\{u_i\} =\{ u_1 , u_2 , u_i , u_m\}$ is the set of objective functions that the players wish to maximize. For every player i, the objective function, u_i, is a function of the particular action chosen by player i, a_i, and the particular actions chosen by all of the other players in the game, a_{-i}. From this model, steady-state conditions, known as *Nash Equilibria* [6] are identified wherein no player would rationally choose to deviate from their chosen action as this would diminish their payoff, i.e. $u_i(a) \le u_i(b_i, a_{-i})$ for all i, j $\in$ M. Nash equilibria model well stables states of a network, since if the network reaches such a configuration, most probably it would remain in the same configuration, since none of the involving entities has a motivation to change his status in order to be more satisfied.

3. The Method

To assess network security, we represent the problem in the form of a *game* between attacking and defending entities [4,5]. When the network designer thinks like an attacker, he/she engages in a game. Finding and evaluating equilibria between attackers' and defenders' strategies provide the means to evaluate the network's security. This information can be provided during the design phase of a prospective network and hence, enables the designer to opt the network features accordingly. Hence, identifying and subsequently evaluating Nash equilibria in prospective networks can help to validate prospective networks' security. However, evaluating security requirements in the design phase, prerequisites the analysis of *all* possible types of assaults for each network's behaviour. These combinations constitute a high number of possible test *scenarios*. Therefore, to evaluate the security performance of a prospective network we need to assess it against each scenario. Scenarios became a popular method for validating NFR [2] where each corresponds to a set of situations that might occur during the operation of a system. Application of scenarios in requirements validation has been performed by a number of researchers [2]. The main problem remaining in requirements validation using scenarios is the specification and subsequently the analysis of a large set of test cases. The large set of scenario variations needed to validate NFRs, overloads the requirements analysis task. On the other hand, automated support for the scenario generation proved to be a vexed problem due to the exponentially large set of possible variations that needs to be examined [2] for the NFR to be guaranteed. An approach that makes this problem tractable is the one described. In particular, we manage to significantly reduce the number of scenarios needed to validate the NFRs by investigating only stable network states (configurations). Actually, our method is of polynomial time complexity compared to the size of the proposed network. Stable configurations describe the most likely states that a network could be in. Thus, by examining network security for only such states, we manage to ensure a satisfactory NFR almost always. Such states are captured through Nash equilibria [6] profiles of the modelled game. Thus, instead of evaluating all possible combinations of network configurations and attacker and defender strategies we only concentrate on Nash equilibria to assess network security.

Our approach is composed of the following three steps:

1. Initially the network designer specifies quantitatively the required level of security he wishes to achieve in the future network.
2. Next, security requirement of the prospective network are modelled in the form of a game using a graph. In particular, we represent the network's topology using a graph and adopt the security game introduced in [4]. Security threats and the potential defence mechanisms are realized using a set of confronting players on a graphical game. It is assumed that the prospective network satisfies some common topological properties. Furthermore, we make some typical assumptions on the attacks that may appear in the network. Moreover it is assumed that we have no prior information on how the attackers behave. Thus, we assume that attacks on the network nodes follow a uniform distribution with equal probability of attacking each node. Game theory is also used to model the specification of the defenders behaviour or mechanisms. This constitutes the functional immunity requirements of the proposed network.
3. Finally, analyse the identified Nash equilibria. For this we use prior knowledge from [4,5] to measure the security guarantee of the prospective network.

4. Application of the Method

First the infrastructural requirements of the prospective network are defined and the required level of the network's security is defined quantitatively. Finding equilibria through Game Theory enables the designer to identify "stable" network configurations that archive the required level of security. This

task is performed analytically. The approach is based on the notion of *scenarios* [2] that correspond to possible configurations of attackers and defenders on the network. The use of Game Theory enables us to reduce the complexity of this process by analysing only scenarios that both attackers and the defender would choose given that they act *rationally* and hence, engage in actions that maximizes their benefit. Through game-theoretic analysis strategies of both attackers and the defenders that maximize their individual benefits are identified. Finding and assessing equlibria among these strategies enables the assessment of prospective network's security. Next section illustrates the application of the method.

4.1 *Network's Topological Specification*

The prospective network N consists of a number of nodes, n, typically representing a set of routers, and a set of communication links E between the nodes of the network. For example consider the network depicted in Figure 1. Moreover, the type of networks we can evaluate must satisfy the following topological property: there exists a subset of links $E' \subseteq E$ such that each node υ of the network belongs to *exactly* one link of the set E'. Network topologies that satisfy this property are called *Matching* networks and can be computationally identified in polynomial time [16]. The network of Figure 1(a) is a Matching network. This is so because the set of edges $E' = \{e_1, e_2, e_3\}$ (thick edges in the Figure) covers all nodes on the network.
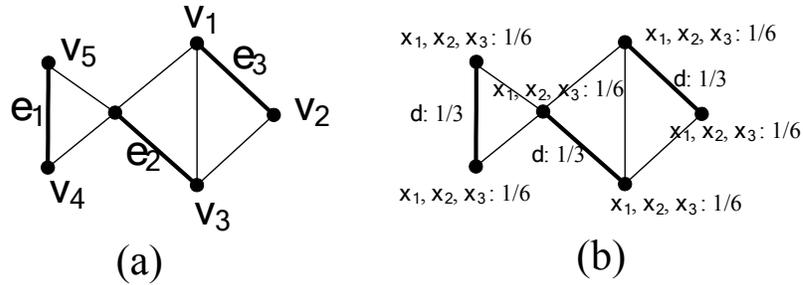

Figure 1: (a) A Matching network. The thick links, edges e_1, e_2, e_3, constitute a Matching set for the network since they hit all nodes of the graph (b) A configuration of a Matching network: The network has 3 attackers, x_1, x_2, x_3, each targeting each node of the network with equal probability 1/6. The defender can be clean edges e_1, e_2, e_3 with equal probability -1/3.

4.2 *Network's Security Specifications*

Network security requirements specification is defined using the common process employed in critical systems specifications. The process consists of the following stages:

a. **Asset identification:** This step addresses the identification of the assets of the network that needs to be protected. In our case, the assets of the network are the nodes of the network. In the most general case, all nodes are of the same importance. A node is considered protected or secure if a security software is installed on that node. Otherwise it is considered vulnerable to attacks.

b. **Threat analysis and assignment:** This step addresses the identification of the possible security threats. These constitute the viruses, worms, Trojan horses and eavesdroppers which are described as attacks that target the nodes of the network. At any time there is a maximum number of attackers, ν, that may be present in the network. Each of them damages nodes that are not protected. In the most general case, we have no information on the distribution of the attacks on the nodes of the network. So, we assume that attacks will follow a uniform distribution, which is quite common in such cases. We call such attacks *uniform attacks*.

c. **Technology analysis:** This step concentrates on the analysis of available security technologies. One major security mechanism for protecting network attacks is the firewalls that we refer to as *defenders*. In distributed firewalls [7] the part of the network protected is defined by the links that the defenders protect. The simplest case is when the sub-network is a single link with its two nodes. In ideal situation the easiest would have been to install software protection in all links of the network. However, due to the financial costs of security software defence mechanisms are only used on a limited part of the network.

d. **Security requirements:** the final step of the security specification process includes the definition of the security requirements that will protect the network against the identified threads. Given the above, in this example we assume that the prospective network will be

supported by a single security mechanism, denoted as *d*, which is capable of cleaning a network link at a time. The position of the defender on the network's nodes is such that satisfies the Matching property.

The assessment and subsequently the validation of security requirements in our method necessitate a game theoretic analysis of the problem. Next, we present the tools necessary to evaluate the security level of the prospective network and finally utilize them in order to validate the security requirement specified by the designer.

4.2.1 *Game Theoretic Modelling*

Network's topological and security specifications are modelled according to sections 4.1 and 4.2 using the graph-theoretic game introduced and investigated in [4,5]. The game is played on a graph *G* representing the network *N*. The players of the game are of two kinds: the *attackers* and the *defender* players. The attackers play on the vertices of the graph, representing the nodes of the network and the defender plays on the edges of the graph, representing the links of the network. The prospective network's *configuration s* of the game is defined by (*i*) the locations of the attackers and (*ii*) the location of the defence mechanism. The positioning attackers and defenders on the network follow a probability distribution that defines the likelihoods of each attacking or defending a node or a link respectively. When attackers target more than one node based on a probability distribution and defenders protect more than one link given another probability distribution, the configuration is defined as *mixed* configuration. Figure 1(b) illustrates a possible configuration for the network. The network satisfies the specifications as defined in Section 4.1. Furthermore, notice that the configuration satisfies the network security specifications of Section 4.2 (A-D): According to this specifications attackers, hit any node on the network uniformly at random, i.e. with probability 1/6, given that the number of nodes is 6. Moreover, the defence mechanism chooses to defend the links of $E' = \{ e_1, e_2, e_3\}$ uniformly at random, where the set E' constitutes a *Perfect Matching, where* the edges of the defenders have no common vertices.

4.2.2 *Security assessment*

To evaluate network security we assess the security level of stable configuration of the game similarly with [4]. Consider a mixed network configuration *s*. Let s_d be the edge selected to being defended by the defender of the resulting respective game defined above. For each attacker $i \in [v]$, let s_i be the node in which the attacker strikes. We say that the attacker *i* is *killed* by the security mechanism if the node s_i is one of the two endpoints of the link s_d being defended by the security software. Then, the *defence ratio* [4] of the configuration *s*, denoted by r_s is defined to be as follows, when given as a percentage:

$$r_s = \frac{\text{expected number of attackers killed in } s}{v} \times 100.$$

Hence, the optimal defence ratio of a network is 100% if the security software manages to kill all attackers. The larger the value of r_s the greater the security level obtained. This approach enables the quantification of security of a perspective network using only examining *stable* configurations. A network whenever reaches a stable a configuration *tents* to remain in the same configuration. This is due to the fact that in such configurations no single player has an incentive to unilaterally deviate from its current strategy. So, such configurations constitute the most probable states of the network. Therefore, we escape from the NP-hard problem of having to assess each possible configuration or scenario. We model stable configuration as Nash equilibria. We evaluate the network security level based on a representative stable configuration. Therefore, through this process we manage to quantify the security level of the prospective network given a *representative* set of stable configurations.

4.2.3 *A Game-Theoretic Validation*

Given specific network configurations the method provides a mechanism to easily estimate the security level of a prospective network using the following theorem.

Theorem 1. [4] *Consider a Matching network N with n nodes and the network security specifications indicated by items* A-D *in section 4.2. Then the network contains a stable configuration s (i.e. a Nash equilibrium) s with level of security given by the following formulae:*

$$r_s = \frac{2}{n} \times 100.$$

Therefore, based on the above, the network of Figure 1(b) has security level equal to 2/6×100=33%, since *n*=6. This designates that the level of security is 33% given the functional requirements specified in configuration *s*. This assessment indicates that the NFR specified by the designer is not satisfied

using the prescribed functional requirements of the network. Hence, the network specification needs to be revised and the security NFR reassessed, prior to proceed to the implementation of the network.

5. Discussion and Conclusion

Security requirements validation is typically performed through security-specific testing. This process is performed in addition to the traditional types of system testing. In this approach, test cases are usually based on abnormal scenarios that describe situations that the network will be called to face. This is analogous to test cases developed for use case based functional testing. These techniques however are mostly based on a reactive paradigm rather than proactive. Moreover, for these to be effective, it is required that a model of the prospective network is developed in a simulator based on which security can be validated. Most importantly, concentrating only on abnormal scenarios limits the effectiveness of the security validation process. Ideally, validation should be performed on all possible scenarios. However, examining all possible scenarios [2] in order to validating security requirements constitutes a highly complex (thus, inefficient) and sometimes infeasible task. In this work we manage to accomplish this process in only polynomial time. This is achieved by considering only stable configurations of the system, that we model using Nash equilibria. In this context, the method presented in this paper constitutes a novelty in validating security NFR through game theory.

The approach presented in this paper is original in security requirements validation since it formally mimics the rationale of the network security problem in a game of attackers and defenders. The application of Game Theory enables the identification of equilibria among the network's defences and the attackers strategies and as a result enables the validation of a prospective networks security NFR using only a limited set of test scenarios. The method usage has been elaborated in a case study that explicitly demonstrates the core steps of the process for validating security requirements. The initial results of this work are encouraging and we are currently looking at techniques to automate the equilibria identification process through the application of systems thinking and system dynamics simulation. Despite its advantages our methods has a number of limitations. Specifically, the assumption that the probability distribution of the attackers' placements on the network is expressed uniformly corresponds to simplified problem scenario. However, there are cases, where prior knowledge of the attackers' behaviour is available. This would lead to different distribution of attacks on the network which subsequently can be utilized to come up with a different defence mechanisms so that to obtain a better security level. Moreover, in our method we assume that only a single defence mechanism is present in the network. However, in large networks usually more than one defence mechanisms are available; although almost always they still can not guarantee absolute network security. As a future work, we plan to utilize other theoretical games that model such scenarios and exploit their analysis in order to provide security requirements validation for more complex networks.

References

[1] R. Crook, D. Ince, L. Lin and B. Nuseibeh, "Security requirements Engineering: When Anti-Requirements Hit the Fan," in *Proceedings of the 10th Anniversary IEEE Joint International Conference on Requirements Engineering*, pp. 203–205, 2002, IEEE Press.

[2] A. Gregoriades and A. Sutcliffe, "Scenario-Based Assessment of Non-Functional Requirements," *IEEE Transactions on Software Engineering*, Vol. 31, no. 5, pp. 392-409, 2005.

[3] A. van Lamsweerde, "Elaborating Security Requirements by Construction of Intentional Anti-Models", in *Proceedings of the 26th International Conference on Software Engineering*, pp. 148–157, 2004, IEEE Press.

[4] M. Mavronicolas, V. G. Papadopoulou, A. Philippou and P. G. Spirakis, "A Network Game with Attacker and Protector Entities," *Algorithmica*, Special Issue with selected papers from the 16th Annual International Symposium on Algorithms and Computation (ISAAC 2005), X. Deng and D. Du guest eds, Vol. 51, No. 3, pp. 315-341, July 2008.

[5] M. Mavronicolas, L. Michael, V. G. Papadopoulou, A. Philippou and P. G. Spirakis, "The Price of Defense", *Proceedings of the 31st International Symposium on Mathematical Foundations of Computer Science,* pp. 717–728, Vol. 4162, Lecture Notes in Computer Science, Springer-Verlag, August/September 2006.

[6] J. F. Nash, ''Non-cooperative Games", *Annals of Mathematics*, 54(2):286–295, 1951.

[7] T. Markham and C. Payne, "Security at the Network Edge: A Distributed Firewall Architecture", in *Proceedings of the 2nd DARPA Information Survivability Conference and Exposition*, Vol. 1, pp. 279–286, June 2001.

[8] D. B. West, *Introduction to Graph Theory*, Prentice Hall, Second edition, 2001.

[9] H. Yuan, G. Chen, J. Wu, H. Xiong, "Towards Controlling Virus Propagation in Information Systems with Point-to Group Information Sharing", *Decision Support Systems,* accepted 20 May 2009.