A Game Theoretic approach based virtual machine migration for cloud environment security

Iman El Mir [1], El Mehdi Kandoussi[1], Mohamed Hanini[1], Abdelkrim Haqiq[1] and Dong Seong Kim[2]

[1]Computer, Networks, Mobility and Modeling laboratory, Hassan 1st Univ, Settat, Morocco
[2]Department of Computer Science and Software Engineering, University of Canterbury, New Zealand

***Abstract*:** In cloud computing environment, static configurations can provide for the attackers an environment too easy for exploitation and discovering the network vulnerabilities in order to compromise the network and launching intrusions; while dynamic reconfiguration seeks to develop a virtual machine (VM) migration over the cloud by applying unpredictability of network configuration's change, and thus improving the system security. In this work a novel approach that performs proactive and reactive measures to ensure a high availability and to minimize the attack surface using VM migration is proposed. This interaction between attack and defense systems was formulated as game model. As result, we have calculated the Nash equilibrium and the utilities for the both attacker and defender, evaluate the parameters which can maximize the defender's utility when the VM migration was planned and identify the potential attack paths. Therefore, the effectiveness of the game model was validated by some numerical results that determine optimal migration strategies in order to ensure the security of the system.

***Keywords*:** Cloud Computing security, VM migration, Attack Graph, Game Theory, Nash equilibrium.

1. Introduction

In the last few years, the cloud computing has emerged as a prominent and widespread paradigm. This technology provides many benefits in terms of flexibility, agility, scalability and saving such as overhead cost reduction due to computing resources optimization. Although these gains offered by this technology, there are also significant obstacles to counter. One of the important obstacles to consider is security. In cloud computing environment, the cloud services on-demand basis are delivered via the Internet. From where, they are facing vulnerabilities and dangers that compromise data confidentiality and availability.

In cloud computing systems, the virtualization technique remains the key component of the cloud where the virtual machines represent the fundamental and core component for the both cloud providers and cloud customers [12]. For cloud providers, VMs maintain the efficiency of the hardware platforms utilization while for cloud customers; VMs minimize the overhead costs due to computing resources maintenance. Moreover, live migration of virtual machines is an essential highlight of virtualization. In such migration, the running VM is transited from one physical host to the next without interrupting the VM and keeping its availability [11], [8]. It's defined as efficient tool for organizations and administrators of cloud data centers due to the better use of physical resources by separating the hardware from the software and making easy fault tolerance, load balancing, and improving manageability. The VM live migration allows high performance to deliver network services and minimizes the service downtimes. These main features of virtual machine live migration can be described as follows:

- **Load balancing**: In the case overload of the physical server, a certain number of VM can be migrated to other PS to minimize the load.
- **Maintenance**: When a physical server needs a maintenance operation that exhibits a downtime of PS, the running VMs on this host should be migrated to others physical machines to continue their operational functioning and availabilities.
- **Power optimization**: the physical servers can be OFF and don't used to minimize the energy consumption, hence, the running VMs on these PS could be moved to others physical machines so as to still available for users.
- **Fault Tolerance**: During the lifecycle of the physical server, this later may fail or malfunctioning. Therefore, the running VMs are migrated to other physical machines, in order to be protected and don't be affected while the dysfunctional server is under-recovering.

Finding a power efficient method to migrate virtual machines from physical server to another physical server in cloud data center is becoming ever more challenging optimization problem in terms of downtime, cost, and performance and particularly in security. Hence, the problem of avoiding vulnerabilities and threats that can be exploited by the attacker to launch a successful attack is a central concern in the security issue in cloud computing environments. Among these attacks, Denial-of-Service (DoS) attack aims to degrade the performance of services by malicious user access; the goal of the internal attacks is to send spiteful code to target VMMs (Virtual Machine Monitor) to harm them. To make the cloud data centers more robust and secure is too difficult and costly. These costs include the cost of vulnerabilities mitigation and attenuation so as to reduce the attack surface and decrease the likelihood to have a successful attack. However, avoidance and discovery of security vulnerabilities in information systems requires awareness of typical risks and a good understanding of vulnerabilities and their exploitation. As solution, by adopting the dynamic computing systems, the attacker cannot have enough time to understand the system configuration and gather full information to launch successful attacks.

To study the mechanism of attack propagation, a number of closely related attack modeling techniques have been developed. Attack graphs are one of such techniques, used to study how an attacker can combine vulnerabilities to stage an attack [34]. Though attack graphs encourages informed risk assessment process and form the basis for optimal network defense, their growth can be exponential and lack the capability to predict attackers set of moves and possible counter-measures. Attack graphs have attracted a considerable attention for modeling systems vulnerabilities and their exploits. However, when the attacker performs network scanning and vulnerability exploiting, the successful exploits may generate a total or partial failure of systems. Attack graphs depict the possible strategies of the attacker to compromise a specific network. By identifying the known vulnerabilities, the attack graphs are automatically generated to counter the important vulnerabilities through proactive recovery or to reconfigure the network so as to prevent the detected attacks, to evaluate the security risk metrics and to discover the shortest attack path.

Network security is a complex and challenging problem. The area of network defense mechanism design is receiving immense attention from the research community for more than two decades. However, the network security problem is far from completely solved. When viewed from a game theoretic perspective, Network security can be seen as a game comprising multiple players; the attackers (malicious users) and the defenders (network/system administrators). The benefit of quantifying network security using game theoretic approach is enormous. Most importantly it may help network administrator to find the optimal defense strategies of a system and to calculate the expected loss associated with different defense strategies [28]. Security games allow a quantitative framework to model the interaction between attackers and defenders. The game models and their solutions are suitable for this aim because they offer a consistent knowledge on attacker's behavior prediction and deploy effective algorithms to make security decision reasonably [4]. In this work, we are exploring the applicability of game theoretic approaches to address the network security issues and some of these approaches look promising. The goal of the research is to design a solution for malicious network attacks using game theory. The challenge of this work is to manage the virtual machine migration in order to mitigate attacks based on attack graph analysis. We have modeled the attack defense strategies using Game Theory approach in a normal form so as to find the Nash equilibrium to evaluate the VM migration benefits and costs. We are exploring the applicability of game theoretic approach to protect the virtual machines during live migration in Cloud Environment. So, the main contributions of this paper can be summarized as follows:

- We have presented our system architecture based on virtual machine migration.
- We have used Mulval tool to generate attack graph corresponding and to enumerate and order all possible attack paths.
- We have modeled the interaction between attacker and defender as a normal game model. The Nash equilibrium of the game is calculated and evaluated to discuss the effectiveness of the VM migration strategy.
- We have validated our proposed game model over some numerical results using Matlab tool.

The rest of the paper is organized as follow: Section 2 provides analysis of some attack-defense strategies with special focus on Game Theory approach in cloud computing environment. The proposed model is described in Section 3. Section 4 presents the theoretical game model for the studied attack-defense schema and its description. Section 5 presents the numerical results and analysis. Finally, section 6 is devoted to the conclusion and future works.

2. Related works

Network security is a complex and challenging issue to be completely solved due to the variety of attack kinds. For example, the advanced persistent threats and zero day attacks are not evident to counter. Hence, networking is a field where the quote that says "easy to attack and hard to defense" [19] can be approved. The attackers can take advantage from time to scan, gather and to exploit the weaknesses and the vulnerabilities of the network systems where the defense system adopts proactive and reactive security mechanisms to address the network security and to find an efficient solution for malicious network attacks. Most current information systems that provide useful services to their legitimate users are connected to the Internet, and it is not obvious to successfully protect such systems against all threats. In this context, various researches have been performed on intrusion tolerance and, multiple defense mechanisms such as intrusion detection system, intrusion prevention system and firewalls have been proposed in order to guarantee high quality of services and to enhance the security.

The authors in [23] have suggested DoS attack detection for server performance improvement without doing any damage to the server. They have proposed Advanced Random Time Queue Blocking with Traffic Prediction (ARTQB- TP) for attack efficiency reduction and attenuation of the impact on server behavior. Through some experimental findings, they have demonstrated the effectiveness of the proposed solution in terms of attenuation of the attack efficiency of the LRDoS attack and service availability to the legitimate user. In [29], the authors suggested a feature that involves the detection of botnet based on statistical approach and comparative study appeared in earlier publications. Their aim is to avert the redundant features, to understand the relationship between influence features so as to minimize the potentials of choosing unnecessary feature which might impact the botnet activity detection. The result carried out exhibit the accuracy reaches about 91% which is approximately admissible to implement the influence feature in detecting botnet activity.

Many tools, models and metrics have been deployed for scanning network vulnerabilities such Nessus [20], but they are limited because only the isolated vulnerabilities are reported while attacker can combine multiple vulnerabilities to penetrate networks. As solution, AGs have been suggested to study the interdependency between security conditions and the vulnerabilities existing in the network [18], [25], [21], [30], [1], [37]. Attack graphs provide a global view of the network in term of network connectivity it gives a detailed analysis of vulnerabilities and their dependencies. Attack graphs can be presented with two distinct manners. In the first possibility, the existing vulnerabilities can be exploited to define all possible attacks paths which attacker can target; however, this can lead to a combinatorial explosion of the numbered attack paths. Secondly, an AG reports the vulnerabilities dependencies and maintains attack paths implicitly so that any information will not be lost. Consequently, attack graphs have a polynomial size calculated by multiplying the number of connected hosts by the number of vulnerabilities existing. A large range of approaches for enumerating network vulnerabilities based on attack graphs such as (Mulval [26], CySeMol [31]), and for filtering and detecting malicious activities for example by IDS alert correlation to locate the attack traces in the generated alerts (as in SnIPS [33] and CRIM [9]) are receiving immense attention from the research community for many years until now. Multiple mathematical approaches and analytical models have been performed to analyze the problems and risks related to security including machine learning [3], data mining [2] and stochastic modeling [17],[14],[13],[15],[16]. The game-theoretic approaches have attracted enormous research attention so as to quantify security. More recently, Game Theory has been used to analyze network security problems and make security decision. In [36], the authors have used non-zero sum game with two scenarios; in DoS attack, one node is considered while with DDoS attack multiple attacking nodes are concerning for modeling the interaction between attacks and defense. The purpose of the defender is to find the optimal firewall to allow accessing only the legitimate traffics while blocking the malicious ones. Their aim is to evaluate the bandwidth consumption in the presence of DoS and DDoS attacks. They have presented two types of game models the static and the dynamic to formulate the Nash equilibrium which represents the best strategy of the defender. They have cross-validated their game model with simulations using NS-3. In [39], the authors implemented the live VM migration to predict the attack through the attack traffic signatures. To consider that the VMs can be heterogeneous, they have proposed the reactive and proactive strategies for Moving Target Defense to gather the heterogeneity for VM pool. In addition, they seek to optimize the cloud resources utilization during migration. In [10], the authors have introduced an MTD architecture which applies the proactive and reactive mechanisms of VM migration so as to promote the cloud based application security and prevent the Denial of Service (DoS) attacks. The challenge of this work is the frequency optimization of migration and the reduction of attack risks. The proposed solution implements the SDN controller based openFlow switches such that when an application is selected for migration to a new virtual machine, all users connected to this application will be redirected to this new target virtual machine. S.Becker and al. [6] have defined a theoretic game framework to determine the best strategy for the both attacker and defender when the defense strategies are well known by the attacker. They have integrated the detection and the control techniques to counter the adversaries. They have proposed to use a decentralized virtual coordinate system which allows creating and keeping a stable set of coordinates that estimate the latency value between nodes without being limited by a fixed infrastructure nodes. For the authors in [24], the aim is to ensure a maximum level of security for all cloud users by grouping users which have the same potential losses at the same hypervisor. They used the term of ”externality” that means the residence of several virtual machines in the same hypervisor but with different levels of security depending on the investments of each user. Consequently, a lack of investment at a virtual machine can create attack risks on the other VMs even if their security levels are high because they reside in a common entity (hypervisor). In this case, they speak of an interdependence which gives rise to a negative externality, otherwise, the term ”positive externality” is used. The problem is that if the attacker reaches the hypervisor and compromise it then all virtual machines running on this hypervisor become compromised and affected. The virtual machines are already allocated in different hypervisors, the idea behind is to choose the hypervisor suited to these virtual machines in term of security. The authors modeled the problem of negative externalities using game theory to analyze all Nash equilibrium for the players so as to minimize the factor externality comparing with other common VM allocation methods. In [22], the use of a common infrastructure between multiple users causes a negative externality for those who invest in security unlike others who do not invest in it. This interdependence in terms of shared resources is one of the common problems that a user of the Cloud environment must take into account in order to decide to invest. In order to model decision-making with respect to security investment by public cloud users, a Game Theory Framework in its normal form has been used. The potential collateral damage incurred by an indirect attack or cross side channel attack is analyzed by the game model. The proposed game model defines multiple possible Nash equilibrium which is dependent on the probability of the hypervisor being compromised at that time. In order to determine the optimal distribution of resources dedicated to detection for a set of virtual machines belonging to a hypervisor, a model based on a zero-sum game is studied in [35]. The authors have proposed a maxmin game with two players the hypervisor and the attacker. Where the hypervisor tries to maximize the detection’s probability, the attacker seeks to minimize this maximization by launching the attack through several VMs.

In a live migration process, the attacker can target the hypervisor and exploits its weaknesses to control and monitor all virtual machines allocated in this hypervisor [32]. Indeed, the data that convey during the migration process are not secure and not encrypted which can be changed or modified by the attacker and the data integrity will be violated [27]. In this regards, many security decisions have been applied. The authors [5] proposed a secure VM migration mechanism. They have proposed Trust-Token to secure the cloud platform and ensure its trustworthiness. The authors in [38] proposed a secure migration process without degradation of protection level. They have used Xen and GNU Linux to implement their scheme and demonstrate the effectiveness of their proposed design. In [7], the authors have used the game theory to predict the stop-and-copy phase so as to minimize the time due to live migration mechanism.

Even if these works have proposed solutions to optimize secure a cloud environment, none of them has exploited, in the proposed schemes, the opportunity given by attack graph to achieve a good compromise between migration and the security issue. By implementing attack graph, we can deeply analyze the system vulnerabilities and threats for decision-making virtual machine migration over the cloud so as to minimize the attacker’s opportunity to launch a successful exploit. The goal of this work is to design a solution for malicious network attacks, using game theory and attack graph, through VM live migration so as to reduce the cost incurred by live migration process, minimize the attack surface and increase the uncertainty for the attacker.

3. System Description and Attack Model

In this section, we describe the system architecture and the attack model implemented to present our proposed VM migration based attack graph analysis. The proposed architecture system depicted in Figure 1 represents Cloud Data Center (CDC) which contains three physical servers $PS_{i\in 1,2,3}$.

On each host, multiple virtual machines are running. In cloud computing environment, the VMs having the same level of security are located in the same class of PS and a migration to a secure PS will mitigate the attack. In the current architecture, PS_1 is considered more secure; the second one PS_2 is less secure than the first while the third PS_3 is not secure. The defender is responsible on managing the dynamic migration strategies to maximize the benefits and to reduce the loss costs. For this reason, we have the system model shown in Figure 2 as a test bed to generate the attack graph described in Figure 2 in order to enumerate all attack strategies and then to evaluate the migration over the cloud. In this context the defender determines the attacker actions by using a black box penetration testing or by using Mulval. This latter generates an exhaustive list of attack paths that aim to compromise a VM as described in Figure 3.

Concerning the attacker, he can be aware of the defender strategies by scanning the network. The result obtained in this recognition phase will be different. Thus, the attacker can conclude that the attack surface of his target changes the configuration and a migration of the targeted VM was occurred when a threat is detected by the defender. This game takes place when the defender knows that an insecure entity is aware of the current configuration of a VM. Practically, the defender detects this threat by continuously analyzing the log of the intrusion detection systems. Therefore, a decision of changing the VM location has to be made. In this security model, the attacker is supposed to be a generic entity that can follow one of the attack path generated by Mulval. Concerning the total loss associated to an attack path, it's calculated by summing different losses after successful executions of rules along the path until reaching the target (in our case it's a VM in a cloud environment).

Concerning the defender, he has three main strategies: he can mitigate an attack by moving the targeted VM to a secure server, to a less secure server or not to move it. The following abbreviations will be used in the rest of this paper as shown in Table 1:

Table 1. Abbreviations

Parameters	Description
$AP_{i\in\{1,...,n\}}$	$AttackPath_{i\in\{1,...,n\}}$
M^S	$SecureMigration$
$M^{\bar{S}}$	$\overline{NotSecure}Migration$
$\overline{M}$	$NoMigration$
VMs	Virtual machines
NE_{pure}	Pure Nash Equilibrium
NE_{mixed}	Mixed Nash Equilibrium

4. Game Model description

4.1 Game formulation

The security game model used in this paper is based on a normal form game. This is a non-cooperative game since any arrangement has been made between the attacker and the defender. Thus, the same steps used in black box penetration testing are used by the attacker and any prior communication will be established by this latter and the defender. The space of actions and the corresponding utility for each outcome are known by all the players. Furthermore, actions are chosen simultaneously and can't be changed during the game. According to the resulting outcome, a reward is attributed differently to each player. This latter is supposed to be rational and pick the action that maximizes his utility.

Formally, a normal form game is defined as follows:

Definition: A strategic game in normal form G is:

- a set I of players ($CardI = n$)
- a set S_i of actions for each player $i \in I$
- an application u from $S = \prod_{i=1}^{n} S_i$ to R^n

$u_i = (s^1, s^2, ..., s^n)$ is the payoff of the player i when the profile $(\mathrm{s}^1, \mathrm{s}^2, ..., \mathrm{s}^n)$ is played.

The set of the players participating in this game is: $I = \{\text{Attacker, defender}\}$. In addition, the set of actions for each player is supposed to be known by the other player. The strategies space of each player is defined as follows (defined by the AG):

$$S_{Attacker} = \{AP_i,\ \mathrm{i} \in (1,...,\mathrm{n})\} \qquad (1)$$

$$S_{Defender} = \{\mathrm{M}^S; \mathrm{M}^{\bar{S}}; \overline{M}\} \qquad (2)$$

Since there are only two players in interaction, a matrix representation has been used. Table 2 shows the utility function for each player according to the profile strategy played. The strategies of the attacker and the defender are represented in columns and in rows respectively. For example if the attacker chooses to follow the AP_i and the defender moves the VM to a secure server, the profile played is: (AP_i, M^s) and the utility for each one is:

$$U_{Defender}(AP_i, M^S) = -\beta_s L_i - C^M \qquad (3)$$

$$U_{Attacker}(\mathrm{AP}_i, \mathrm{M}^S) = \beta_s L_i - C_i \qquad (4)$$

These expressions and used parameters are explained below. In case of a secure migration, the probability to identify again the targeted VM is β_s. The same case occurs when the destination server is less secure than the current server and, the probability is denoted $\beta_{\bar{s}}$ with $\beta_s < \beta_{\bar{s}}$. In addition a migration has in general a cost in terms of time of availability and bandwidth consumption. Therefore, to minimize these two parameters, an alternative solution is used. Indeed to minimize the time of unavailability, a second image of the VM is created. So in all of the cases, the process of migration has a cost C^M and in general it is very low comparing to the loss of an attack. Each attack path

AP_i has an estimated loss L_i (for the defender) and an estimated cost C_i (for the attacker). The cost C_i of an attack is calculated based on time consumed in the recognition phase and in the use of computational requirements as brute force. The hypotheses of this game are summarized as follows:

- $\forall i \in \{1,...,\mathrm{n}\}\ \mathrm{L}_i > 0$ and $\mathrm{C}_i > 0$
- Without loss of generality, the order of the AP_i in the matrix representation is : $L_n > ... > L_1 > C^M > 0$
- $0 < \beta_s < \beta_{\bar{s}} < 1$

Assuming that the attacker chooses AP_i and no migration has been occurred. In this case this VM will be compromised and the utility of the attacker will be $L_i - C_i$. This latter is composed of two parts: A negative part called cost of the attack C_i and a positive part L_i representing the loss. These two values are intrinsic to the chosen attack path. In the other side, the defender obtains a negative utility equal to $-L_i$. In the case where a secure migration is chosen, the utility of the attacker will be $\beta_s L_i - C_i$. The quantity $\beta_s L_i$ represents the expected loss of the VM. The defender utility is $-\beta_s L_i - C^M$. This latter is an sum of the migration cost and $-\beta_s L_i$. A similar explanation can be done in the case where a migration to a less secure server was happened.

4.2 Game resolution

Resolving a normal form game is by finding Nash equilibrium (or a set of Nash equilibria). In the following, instead of writing $s = (s_1, ..., \mathrm{s}_n)$, the notation $s = (s_i, s_{-i})$ is used. s_{-i} denotes the actions vector of the players other than i.

***Definition* (Best response):** A strategy s_i^* is denoted as a best response of the player i if:

$$\forall s_i \in S_i,\ \mathrm{u}_i(s_i^*, s_{-i}) \geq \mathrm{u}_i(\mathrm{s}_i, \mathrm{s}_{-i}) \tag{5}$$

***Definition* (Strictly dominated strategy)**: A strategy $s_i \in S_i$ of the player i is strictly dominated by $s_i^{'}$ if:

$$\forall s_{-i} \in S_i, u_i(s_i, s_{-i}) < u_i(\mathrm{s}_i^{'}, \mathrm{s}_{-i}) \tag{6}$$

Definition (Nash equilibrium): The profile strategy $s^* = \left\langle s_1^*, ..., \mathrm{s}_n^* \right\rangle$ is a Nash Equilibrium if:

$$\forall i \in I, s_i \in S_i : u_i(\mathrm{s}_i^*, \mathrm{s}_{-i}^*) \geq u_i(\mathrm{s}_i, \mathrm{s}_{-i}^*) \tag{7}$$

According to the hypothesis $\beta_s < \beta_{\bar{s}}$, the strategy $M^{\bar{s}}$ is strictly dominated by M^s. Therefore, the defender strategies space will be reduced to:

$$S_{Defender}^{'} = \{\overline{M}, \mathrm{M}^s\} \tag{8}$$

We consider these two subsets of the attacker strategies space:

$$E = \{\mathrm{AP}_i : \exists\, \mathrm{j} > \mathrm{i},\ \mathrm{C}_j - \mathrm{C}_i \leq 0\} \tag{9}$$

$$F = \{\mathrm{AP}_j : \exists \mathrm{i} < \mathrm{j}, \frac{C_j - C_i}{L_j - L_i} \geq 1\} \tag{10}$$

Given a strategy $AP_i \in \mathrm{S}_{Attacker}$ if $AP_i \in \mathrm{E}$ or $AP_i \in \mathrm{F}$. This strategy is strictly dominated by another $AP_{i^{'}}$ where $i \neq i^{'}$. Therefore, the attacker strategy space will be reduced to:

$$S_{Attacker}^{'} = S_{Attacker} \setminus \{\mathrm{E} \cup \mathrm{F}\} \tag{11}$$

Assuming $m = \mathrm{Card}(\mathrm{S}_{Attacker}^{'})$ and $\alpha_{j,i} = \dfrac{C_j - C_i}{L_j - L_i}$ where $j > i$, for each two attack paths i et j where $j > i$ from $S_{Attacker}^{'}$, the following inequality is verified: $0 < \alpha_{j,i} < 1$.

Moreover, the elements of the set $A = \{\alpha_{j,i} \,/\, \mathrm{j} > \mathrm{i}\}$ are ordered as follows:

$$0 < ... < \alpha_{j,i} < \alpha_{j^{'},i^{'}} < ... < 1 \tag{12}$$

Given $\beta_s \in \left]0,1\right[$

$\exists! \alpha_{j,i} : \beta_s \in \left[\alpha_{j,i}, \alpha_{j^{'},i^{'}}\right[\cup \left[0, \min A\right[\cup \left[\min A, 1\right]$

Therefore,

$$\exists! i_0 \in \{1,...,\mathrm{m}\} \forall \mathrm{i} \neq \mathrm{i}_0, -\mathrm{C}_{i_0} + \beta_s L_{i_0} > -\mathrm{C}_i + \beta_s L_i \tag{13}$$

Noting $\Delta(S_{Attacker}^{'})$ (resp. $\Delta(S_{Defender}^{'})$) the set of the probability distribution over the attacker strategy space (resp. the defender strategy space) and given $\lambda \in \Delta(\mathrm{S}^{'})$, $Supp(\lambda)$ is the set of elements that have a non-null probability over a given set. For a fixed value of β_s :

– if M^s is played by the defender, the best response of the attacker is AP_{i_0}. According to equation (13), AP_{i_0} maximizes the attacker's utility.

– if M is played by the defender, the best response of the attacker is AP_m .

Therefore,

$$Supp(\lambda_{Attacker}) \subset \{\{\mathrm{AP}_{i_0}\}, \{AP_m\}, \{\mathrm{AP}_{i_0}, AP_m\}\} \tag{14}$$

Table 2. Payoff Matrix of the game

	$\cdots$	$AttackPath_i$	$\cdots$	$AttackPath_n$
SecureMigration		$-\beta_s L_i - C^M$ $\beta_s L_i - C_i$		$-\beta_s L_n - C^M$ $\beta_s L_n - C_n$
NoMigration		$-L_i$ $L_i - C_i$		$-L_n$ $L_n - C_n$
$\overline{Secure}Migration$		$-\beta_{\bar{s}} L_i - C^M$ $\beta_{\bar{s}} L_i - C_i$		$-\beta_{\bar{s}} L_n - C^M$ $\beta_{\bar{s}} L_n - C_n$

- if $i_0 = m$:

- if $\beta_s < 1 - \dfrac{C^M}{L_m}$: $\mathrm{NE}_{pure} = (\mathrm{M}^s, \mathrm{AP}_m)$ then,
  - $U_{\mathrm{Defender}} = -\beta_s . L_m - C^M$

  - ○ $U_{\text{Attacker}} = -C_m + \beta_s . L_m$

- if $\beta_s \geq 1 - \frac{C^M}{L_m}$: $\mathrm{NE}_{pure} = (\overline{\mathrm{M}}, \mathrm{AP}_m)$ then,
  - ○ $U_{\text{Defender}} = -L_m$
  - ○ $U_{\text{Attacker}} = -C_m + \mathrm{L}_m$

- if $i_0 \neq m$:

- if $\beta_s \leq 1 - \frac{C^M}{L_{i_0}}$: $\mathrm{NE}_{pure} = (\mathrm{M}_s, \mathrm{AP}_{i_0})$ then,
  - ○ $U_{\text{Defender}} = -\beta_s L_{i_0} - C^M$
  - ○ $U_{\text{Attacker}} = -C_{i_0} + \beta_s L_{i_0}$
- if $\beta_s \geq 1 - \frac{C^M}{L_m}$: $\mathrm{NE}_{pure} = (\overline{M}, \mathrm{AP}_m)$ then,
  - ○ $U_{\text{Defender}} = -L_m$
  - ○ $U_{\text{Attacker}} = -C_m + L_m$
- If $1 - \frac{C^M}{L_{i_0}} < \beta_s < 1 - \frac{C^M}{L_m}$: there is a mixed Nash

equilibrium:

$$NE_{mixed} = (\mathrm{x} . \mathrm{AP}_{i_0} + (1-x).AP_m ; y.M^s + (1-y).\overline{M}) \quad (15)$$

where

$$x = \frac{L_m(1-\beta_s) - C_M}{(L_m - L_{i_0})(1-\beta_s)} \text{ and } \mathrm{y} = \frac{1}{1-\beta_s} . \left(1 - \frac{C_m - C_{i_0}}{L_m - L_{i_0}}\right) \quad (16)$$

then,

$$\begin{aligned} U_{\text{Defender}} &= -x.\left[y.(\beta_s L_{i_0} + C^M) + (1-y).L_{i_0} \right] \\ &\quad - (1-x).\left[y.(\beta_s L_m + C^M) + (1-y).L_m \right] \end{aligned}$$

and

$$\begin{aligned} U_{\text{Attacker}} &= x.\left[y.(\beta_s L_{i_0} - C_{i_0}) + (1-y).(L_{i_0} - C_{i_0}) \right] \\ &\quad + (1-x).\left[y.(\beta_s L_m - C_m) + (1-y).(L_m - C_m) \right] \end{aligned}$$

Indeed, at the mixed Nash equilibrium, the attacker must randomize in order to make the defender indifferent to choosing either strategy. Explicitly this approach is translated by the equation:

$$EU_1(\mathrm{M}^s) = \mathrm{EU}_1(\overline{M}) \Leftrightarrow x = \frac{L_m(1-\beta_s) - C_M}{(L_m - L_{i_0})(1-\beta_s)} \quad (17)$$

By applying the same approach to the defender situation, we get:

$$EU_2(\mathrm{AP}_m) = \mathrm{EU}_2(AP_{i_0}) \Leftrightarrow y = \frac{1}{1-\beta_s} . \left(1 - \frac{C_m - C_{i_0}}{L_m - L_{i_0}}\right) \quad (18)$$

5. Numerical results

Game analysis provides an in-depth explanation of pure and mixed Nash equilibrium. In this section only three attack paths are considered. Indeed, the variables used thereafter are: L_1 , L_2 , L_3 , C_1 , C_2 , C_3 , C^M and β_s. The variable $\beta_{\bar{s}}$ will be always considered greater than β_s and no value will be assigned to it. According to The values of the variables cited above are taken in the same order as that in [24][22]: L_1 = 250, L_2 = 1100, L_3 = 1500, C_1 = 50, C_2 = 350, C_3 = 650 and C^M = 300.

In figures 4 and 6, distributions of probability over the actions of the defender and the attacker in case of equilibrium are illustrated. When such equilibrium occurs no player has any incentive to individually deviate their strategy. This situation must optimize both the defender's and the attacker's models at the same time, which means that both the attacker's and defender's best response functions should be satisfied at this point.

In Figure 4, the probability distribution over attack paths in Nash Equilibrium are represented below with respect to the likelihood of the VM identification after migration: β_s. According to 4.a, the attack path AP_1 is the best response of the attacker when $\beta_s \leq 0.35$. Indeed, this is due to the complexity of identifying the VM after a migration and the attacker has the maximum utility even if the AP_1 has the less loss L_1.

For values of $\beta_s : 0.35 \leq \beta_s \leq 0.50$ (resp. $0.70 \leq \beta_s \leq 0.75$), we have a mixed Nash Equilibrium with $Supp = \{AP_1, AP_3\}$ (resp. $Supp = \{AP_2, AP_3\}$) illustrated in 4.a, 4.b and 4.c. In the same way AP_2 is a best response when $0.50 \leq \beta_s \leq 0.70$. For higher value of β_s the attacker follows the AP_3 which has a large loss on the targeted VM. We can conclude that the parameter β_s gives an idea about the potential attack path and helps the defender to define the optimal measure of security to take in terms of the best server to choose for hosting the migrated VM.

The attacker's utility with respect to β_s is depicted in Figure 5. This utility increases with β_s. This is intuitive because the VM can't be exploited by the attacker only if it is easily identifiable. In addition, for values of $\beta_s \leq 0.10$, attacker shouldn't follow any attack path. Indeed, in this interval his utility is negative. Thus, the VM is secure when $\beta_s \leq 0.10$ and any attempt to exploit this latter will affect negatively the utility of the attacker. In other point of view, the value $\beta_s = 0.1$ gives an idea about the destination server in order to mitigate totally the attack. For values of $\beta_s \geq 0.79$, attacker has a constant utility. This corresponds to the attack path AP_3 providing the greatest potential loss.

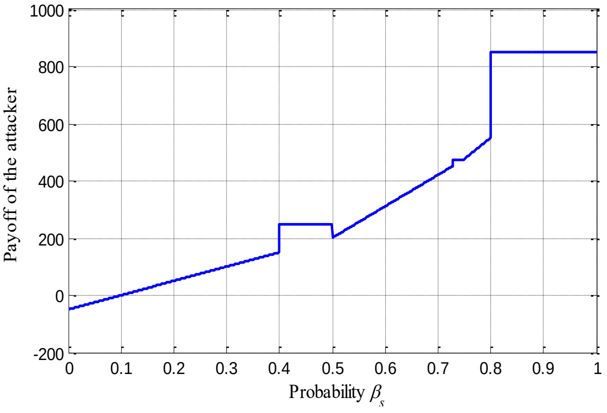


Figure 5. Payoff of the attacker with respect to β_s

In Figure 6, the probability distribution over $\{M^s, \bar{M}\}$ in Nash Equilibrium are represented with respect to the likelihood of the VM identification after migration: β_s. Concerning the migration to a less secure server ($M^{\bar{s}}$ strategy), we have demonstrated that it can't be played by the defender since it's a dominated strategy. Only distribution over $\{M^s, \bar{M}\}$ are illustrated. According to Figure 6, the migration must be carried out in general if β_s is lower than a certain threshold: $\beta_s \leq 0.79$. Otherwise the defender has to choose not to migrate the VM. Indeed, as depicted below, for value of $0.35 \leq \beta_s \leq 0.50$ or $0.70 \leq \beta_s \leq 0.75$ the defender has to randomly to choose between M^s or $\bar{M}$ and it's preferable to migrate the VM since $\Pr(M^s) > \Pr(\bar{M})$. From another point of view, this threshold gives an idea of the destination servers candidate to host the migrated VM. For servers having a high value $\beta_s \geq 0.79$ the VM needn't to be migrated.

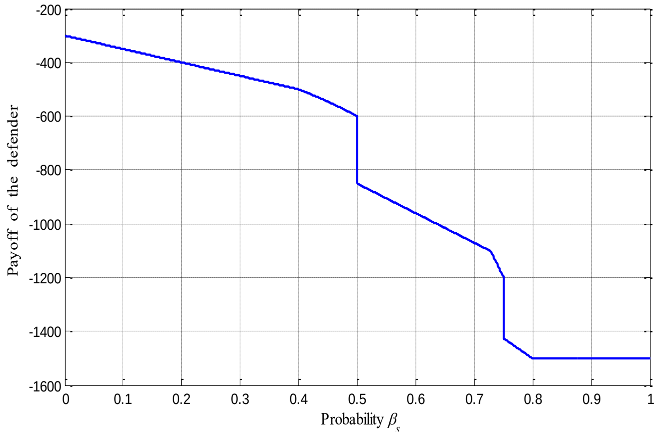


Figure 7. Payoff of the defender with respect to β_s

The defender's utility is depicted in Figure 7 with respect to β_s.It is clear that the defender's utility decreases with the probability β_s. This is intuitive, because when β_s increases the VM will not be secure and can provide enough information to be identifiable in the destination server. As shown in the Figure 7, the utility function of the defender is composed from three parts (three discontinuities in general): the first part corresponds to the value of $\beta_s \leq 0.5$. In this case the attacker uses the attack path AP_1 which has the minimum loss compared to the other two. And this is translated by the fact that the defender's utility has higher values in this part. The same explanation goes for the values $0.5 \leq \beta_s \leq 0.75$. Concerning the last part of the utility function when $\beta_s \geq 0.75$ we have a minimum defender's utility and in this case the VM is not secure and this is due also to the AP_3 which has the higher negative effect on the VM since $L_3 \succ L_2 \succ L_1$. The variation of the defender's payoff gives the idea about the average loss in case of a secure migration and the source of the potential threat.

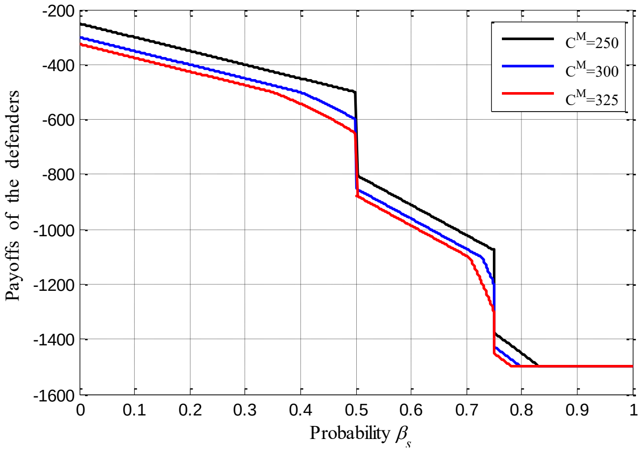


Figure 8. Payoff of the defender with respect to β_s and C^M

In Figure 8, the defender's utility is represented with respect to the cost of migration C^M and β_s. As illustrated, the utility of the defender decreases considerably with the increase of C^M and β_s. For values higher than the threshold: $\beta_s \geq 0.79$ it is clear that investment at the level of migration does not have any impact on utility. Indeed, only security measures related to the likelihood of identification must be taken into consideration.

6. Conclusion and Future work

In this paper, we proposed a game theory model to analyze the interaction between service provider and attacker, and to optimize the security in a cloud computing environment. In this model the defender manages a homogeneous VM pool with the strategies to migrate virtual machines from one physical server to another in the cloud center so as to increase the attack uncertainty and to minimize the attack surface. A reactive and proactive VM migration taking into consideration the probability of identification of the VM in case of its migration is considered. The strategies of the attacker are deduced from an attack graph that enumerates all possible attacks. The mathematical analysis and the numerical results showed that the proposed model can help to predict the potential attack path that can compromise the VM, to determine in which case the VM should be migrated, and to select the destination server. As future work, we seek to mitigate the attacks in cloud computing environment by using a cooperative game and the load over VMs hosted on a hypervisor will be combined to our model based on migration.

References

[1] J.C. Acosta, E. Padilla, J. Homer, "Augmenting attack graphs to represent data link and network layer vulnerabilities", Military Communications Conference, MILCOM IEEE. Baltimore, MD, USA, pp. 1010-1015, 2016.

[2] A.O. Adetunmbi, S.O. Falaki, O.S. Adewale, B.K. Alese, "Network intrusion detection based on rough set and k-nearest neighbor", International Journal of Computing and ICT Research , Vol. 2, No. 1, pp. 60–66, 2008.

[3] A.O. Adetunmbi, B.K. Alese, O. Ogundele, S.O. Falaki, "A data mining approach to network intrusion detection", Journal of Computer Science & Its Application, Vol. 1, No. 2, pp. 24-37, 2007.

[4] T. Alpcan, T. Basar," Network security: A decision and game-theoretic approach", Cambridge University Press, 2010.

[5] M. Aslam, C. Gehrmann, M. Bjorkman, "Security and trust preserving vm migrations in public clouds", IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, Liverpool, United Kingdom, pp. 869–876, 2012.

[6] S. Beckery, J. Seibert, D. Zage, C. Nita-Rotaru, R. Statey, "Applying game theory to analyze attacks and defenses in virtual coordinate systems", 41st International Conference on Dependable Systems & Networks (DSN), Hong Kong, China, pp. 133–144, 2011.

[7] Y.L. Chen, Y.C. Yang, W.T. Lee, "The study of using game theory for live migration prediction over cloud computing", Springer, Intelligent Data analysis and its Applications, Vol. 2 , pp. 417–425, 2014.

[8] C. Clark, K. Fraser, S. Hand, J.G. Hansen, E. Jul, C. Limpach, I. Pratt, A. Warfield, "Live migration of virtual machines", The 2nd Conference on Symposium on Networked Systems Design & Implementation, Boston, Massachusetts, USA, Vol. 2, pp. 273–286, 2005.

[9] F. Cuppens, A. Miege, "Alert correlation in a cooperative intrusion detection framework", IEE symposium Security and privacy, Oakland, California, USA, pp. 202–215, 2002.

[10] S. Debroy, P. Calyam, M. Nguyen, A. Stage, V. Georgiev, "Frequency-minimal moving target defense using software-defined networking", International Conference IEEE Computing, Networking and Communications (ICNC), New York, NY, USA, pp. 1–6, 2016.

[11] R. Divyambika, A. Umamakeswari, "Protection of virtual machines during live migration in cloud environment", Indian Journal of Science and Technology 8(S9), pp. 333–339, 2015.

[12] S. El Kafhali, K. Salah, " Stochastic modelling and analysis of cloud computing data center", 20th ICIN Conference Innovations in Clouds, Internet and Networks. Paris, France, pp. 122–126, 2017.

[13] I. El Mir, A. Chowdhary, D. Huang, S. Pisharody, D.S. Kim, A. Haqiq, "Software defined stochastic model for moving target defense", Third International Afro-European Conference for Industrial Advancement (AECIA16) Springer, Marrakesh, Morocco, p. 188-197, 2016.

[14] I. El Mir, A. Haqiq, D.S. Kim, "Performance analysis and security based on intrusion detection and prevention systems in cloud data centers", Springer International Conference on Hybrid Intelligent Systems, Marrakesh, Morocco, pp. 456–465, 2016.

[15] I. El Mir, D.S. Kim, A. Haqiq, "Security modeling and analysis of a self-cleansing intrusion tolerance technique", 11th International Conference Information Assurance and Security (IAS), Marrakesh, Morocco, pp. 111–117, 2015.

[16] I. El Mir, D.S. Kim, A. Haqiq, "Security modeling and analysis of an intrusion tolerant cloud data center", Third World Conference Complex Systems (WCCS), Marrakesh, Morocco, pp. 1–6, 2015.

[17] I. El Mir, D.S. Kim, A. Haqiq, "Cloud computing security modeling and analysis based on a self-cleansing intrusion tolerance technique", Journal of Information Assurance & Security, Vol. 11, No. 5, 2016.

[18] H. Holm, T. Sommestad,: "Sved: Scanning, vulnerabilities, exploits and detection", Military Communications Conference MILCOM, Baltimore, MD, USA, pp. 976–981, 2016.

[19] S. Jajodia, A.K. Ghosh, V. Swarup, C. Wang, X.S. Wang, "Moving target defense: creating asymmetric uncertainty for cyber threats", Springer Science & Business Media, Vol. 54, 2011.

[20] S. Jajodia, S. Noel, B. OBerry, "Topological analysis of network attack vulnerability", Springer Managing Cyber Threats, pp. 247–266, 2005.

[21] P. Johnson, A. Vernotte, M. Ekstedt, R. Lagerstrom, "pwnpr3d: An attack-graph-driven probabilistic threat-modeling approach", 11th International Conference Availability, Reliability and Security (ARES), Salzburg, Austria, pp. 278–283, 2016.

[22] C.A. Kamhoua, L. Kwiat, K.A. Kwiat, J.S. Park, M. Zhao, M. Rodriguez, "Game theoretic modeling of security and interdependency in a public cloud", 7th International Conference Cloud Computing (CLOUD), Anchorage, AK, USA, pp. 514–521, 2014.

[23] R. Kavitha, and G. Padmavathi, "Advanced Random Time Queue Blocking with Traffic Prediction for Defense of Low-rate DoS Attacks against Application Servers", *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 9, No. 1, pp. 95-104, 2017.

[24] L.Kwiat, C.A. Kamhoua, K.A. Kwiat, J. Tang, A. Martin, "Security-aware virtual machine allocation in the cloud: A game theoretic approach", 8th International Conference Cloud Computing (CLOUD), New York, NY, USA, pp. 556–563, 2015.

[25] H.H. Nguyen, K. Palani, D.M. Nicol, "An approach to incorporating uncertainty in network security analysis", The Hot Topics in Science of Security: Symposium and Bootcamp, Hanover, MD, USA, pp. 74–84, 2017.

[26] X. Ou, S. Govindavajhala, A.W. Appel, "Mulval: A logic-based network security analyzer", USENIX security symposium, Baltimore, MD, USA, pp. 8, 2005.

[27] S.B. Rathod, V.K. Reddy, "Secure live vm migration in cloud computing: A survey", International Journal of Computer Applications, Vol. 103, No .2, 2014.

[28] K. Sallhammar, S.J. Knapskog, B.E. Helvik, "Using stochastic game theory to compute the expected behavior of attackers", The 2005 Symposium on Applications and the Internet Workshops, Trento, Italy, pp. 102–105, 2005.

[29] N.H.M. Saudi,"Revealing the Feature Influence in HTTP Botnet Detection", *International Journal of Communication Networks and Information Security (IJCNIS)*, 2017, vol. 9, No. 2, pp. 274-281, 2017.

[30] A. Sen, S. Madria, "Risk assessment in a sensor cloud framework using attack graphs", IEEE Transactions on Services Computing, Vol. pp, No. 99, pp. 1-1, 2016.

[31] T. Sommestad, M. Ekstedt, H. Holm, "The cyber security modeling language: A tool for assessing the vulnerability of enterprise system architectures", IEEE Systems Journal, Vol. 7, No. 3, pp. 363– 373, 2013.

[32] D. Sun, J. Zhang, W. Fan, T. Wang, C. Liu, W, Huang, "Splm: security protection of live virtual machine migration in cloud computing", The 4th ACM International Workshop on Security in Cloud Computing, Xi'an, China, pp. 2–9, 2016.

[33] S.C. Sundaramurthy, L. Zomlot, X. Ou, ”Practical ids alert correlation in the face of dynamic threats”, International Conference on Security and Management (SAM11), LasVegas, USA, 2011.

[34] R. Trost, “Practical intrusion analysis: prevention and detection for the twenty-first century”, Pearson Education, 2009.

[35] O.A. Wahab, J. Bentahar, H. Otrok, A. Mourad, “How to distribute the detection load among virtual machines to maximize the detection of distributed attacks in the cloud? “, IEEE International Conference Services Computing (SCC), San Francisco, USA, pp. 316–323, 2016.

[36] Q. Wu, S. Shiva, S. Roy, C. Ellis, , V. Datla, “On modeling and simulation of game theory based defense mechanisms against dos and ddos attacks”, The spring simulation multiconference. San Diego, CA, USA, p. 159, 2010.

[37] R. Yadav, R.N. Verma, A.K. Solanki, ”An improved model for analysis of host network vulnerability”, International Journal of Computer Applications, Vol. 148, No. 13, pp. 12-16, 2016.

[38] F. Zhang, Y. Huang, H. Wang, H. Chen, B. Zang, “Palm: security preserving vm live migration for systems with vmm-enforced protection”, The Third Asia-Pacific Trusted Infrastructure Technologies Conference, Hubei, China, pp. 9–18, 2008.

[39] R. Zhuang, S. Zhang, A. Bardas, S.A. DeLoach, X. Ou, A. Singhal, “Investigating the application of moving target defenses to network security”, 6th International Symposium Resilient Control Systems (ISRCS), San Francisco, CA, USA, pp. 162–169, 2013.

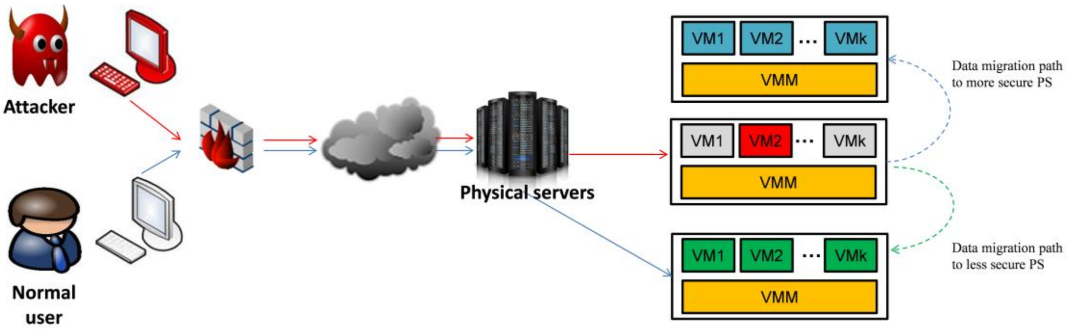

Attacker
Normal user
Physical servers
VM1
VM2
VMk
VMM
Data migration path to more secure PS
Data migration path to less secure PS

Figure 1. System architecture description based VM migration techniques to prevent malicious attacks.

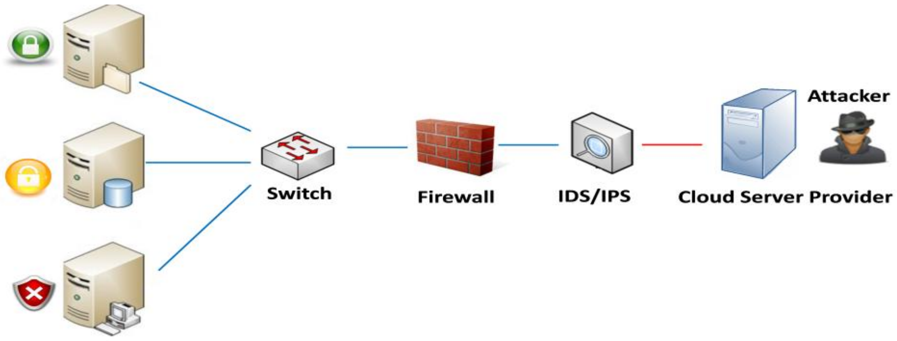
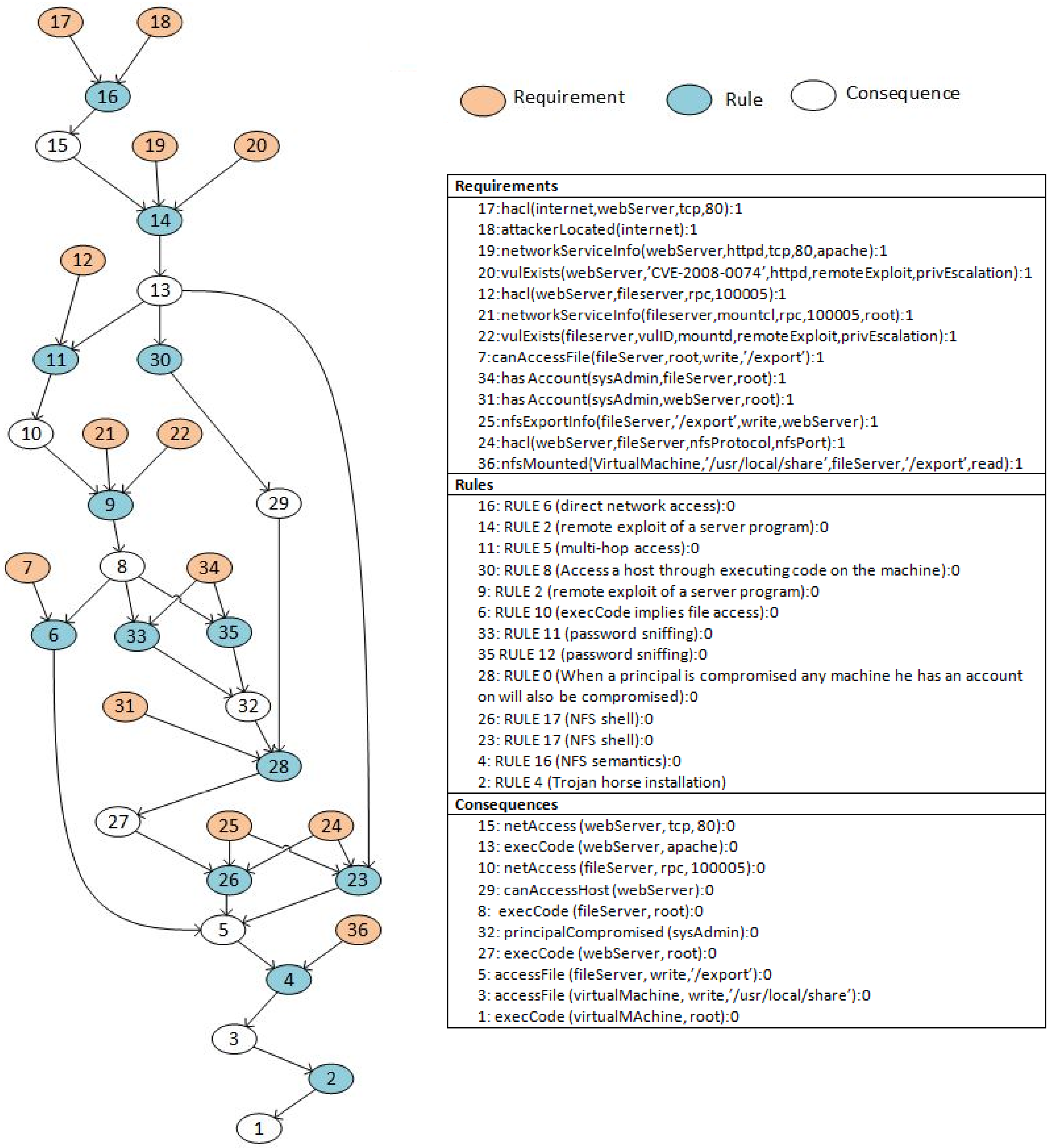


Figure 2. System Model illustration



Requirements

17:hacl(internet,webServer,tcp,80):1
18:attackerLocated(internet):1
19:networkServiceInfo(webServer,httpd,tcp,80,apache):1
20:vulExists(webServer,'CVE-2008-0074',httpd,remoteExploit,privEscalation):1
12:hacl(webServer,fileserver,rpc,100005):1
21:networkServiceInfo(fileserver,mountcl,rpc,100005,root):1
22:vulExists(fileserver,vulID,mountd,remoteExploit,privEscalation):1
7:canAccessFile(fileServer,root,write,'/export'):1
34:has Account(sysAdmin,fileServer,root):1
31:has Account(sysAdmin,webServer,root):1
25:nfsExportInfo(fileServer,'/export',write,webServer):1
24:hacl(webServer,fileServer,nfsProtocol,nfsPort):1
36:nfsMounted(VirtualMachine,'/usr/local/share',fileServer,'/export',read):1

Rules

16: RULE 6 (direct network access):0
14: RULE 2 (remote exploit of a server program):0
11: RULE 5 (multi-hop access):0
30: RULE 8 (Access a host through executing code on the machine):0
9: RULE 2 (remote exploit of a server program):0
6: RULE 10 (execCode implies file access):0
33: RULE 11 (password sniffing):0
35 RULE 12 (password sniffing):0
28: RULE 0 (When a principal is compromised any machine he has an account on will also be compromised):0
26: RULE 17 (NFS shell):0
23: RULE 17 (NFS shell):0
4: RULE 16 (NFS semantics):0
2: RULE 4 (Trojan horse installation)

Consequences

15: netAccess (webServer, tcp, 80):0
13: execCode (webServer, apache):0
10: netAccess (fileServer, rpc, 100005):0
29: canAccessHost (webServer):0
8: execCode (fileServer, root):0
32: principalCompromised (sysAdmin):0
27: execCode (webServer, root):0
5: accessFile (fileServer, write,'/export'):0
3: accessFile (virtualMachine, write,'/usr/local/share'):0
1: execCode (virtualMAchine, root):0

Figure 3. Attack graph generated from MulVAL

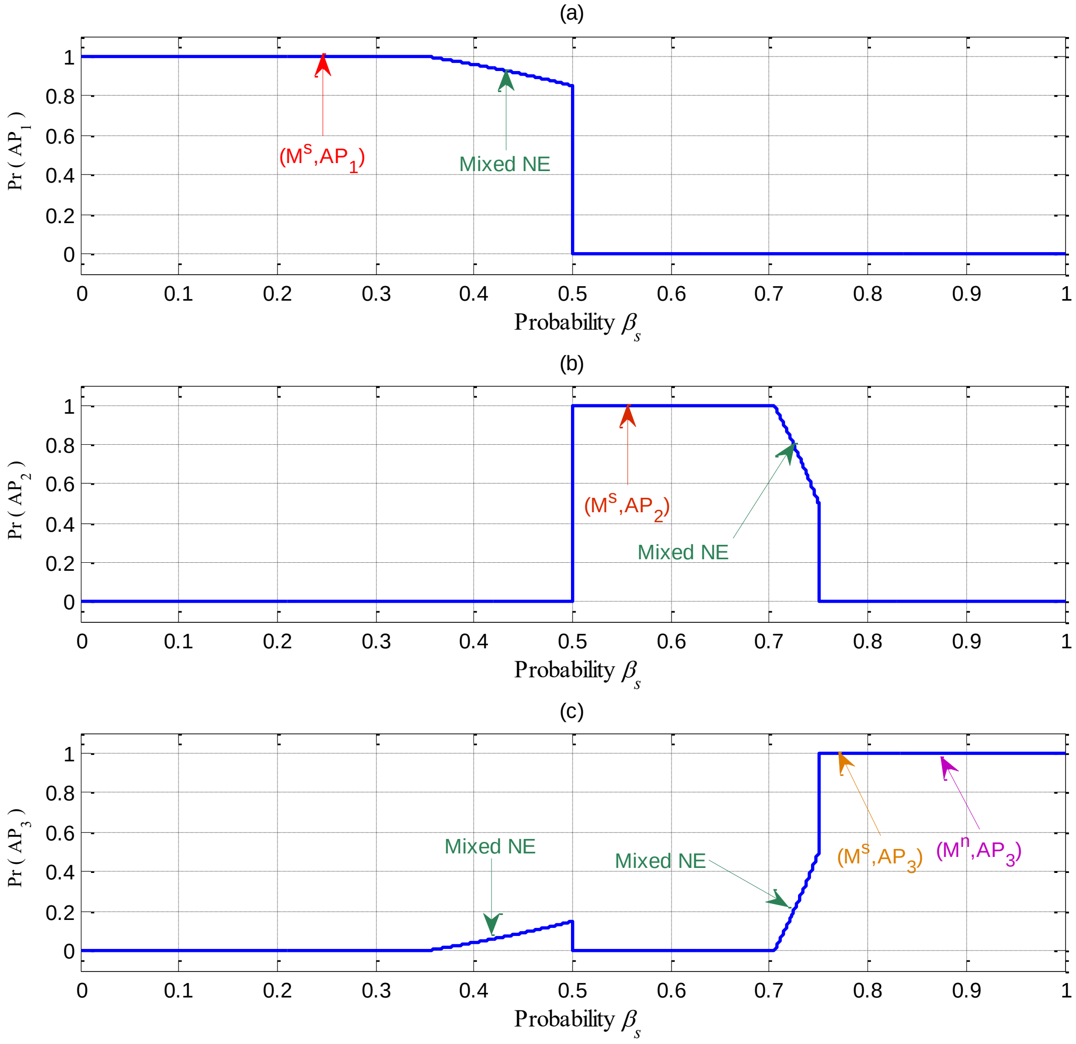


Figure 4. Probability distribution over attack paths

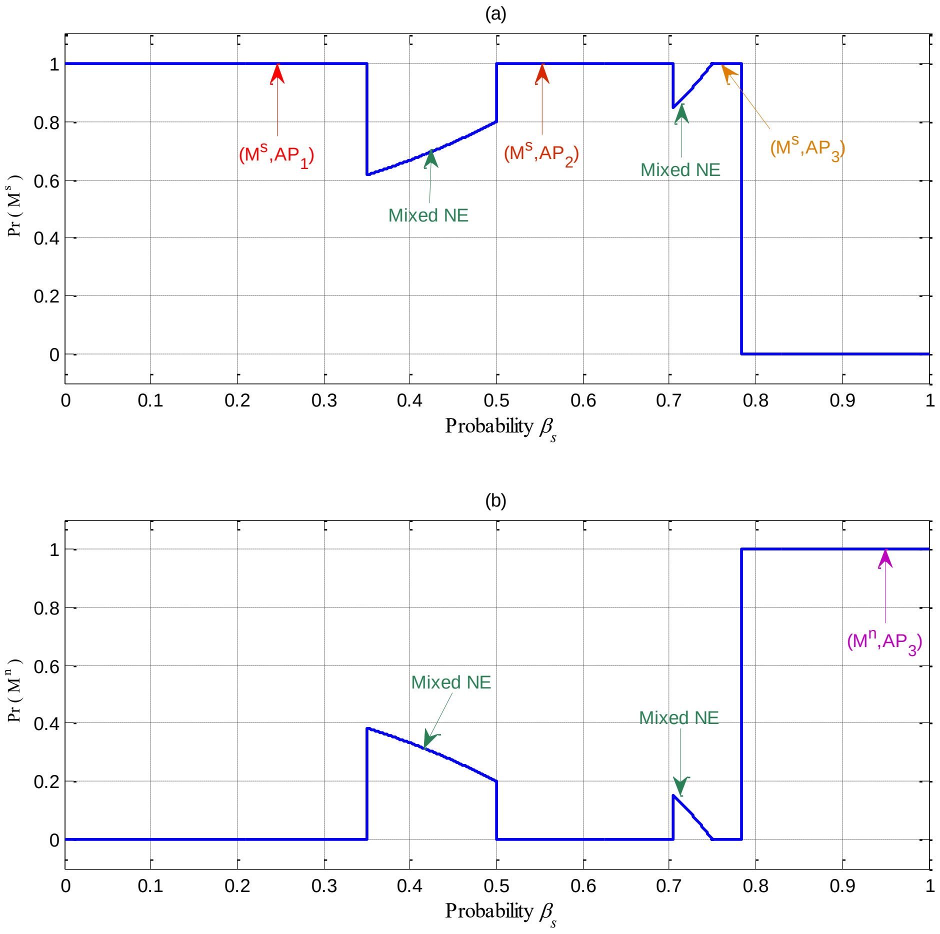


Figure 6. Probability distribution over defender actions