A Model for Analyzing the Effect of Moving Target Defenses on Enterprise Networks

Rui Zhuang Scott A. DeLoach Xinming Ou
Kansas State University
Manhattan, KS USA
{zrui, sdeloach, xou}@ksu.edu

ABSTRACT

This paper presents an analytical model for determining the effectiveness of moving target defense (MTD) systems in an enterprise network environment. The goal of our model is not to predict the exact probabilities involved with a MTD system, but to provide insight to designers that allows them to make better design decisions when designing their enterprise networks. We validate the model using a simulation-based of attackers and the MTD system.

Categories and Subject Descriptors

K.6.5 [**Security and Protection**]: Unauthorized access—*Management of computing and information system*

Keywords

moving target defense, enterprise network security

1. INTRODUCTION

Currently, enterprise network configurations tend to be static, which gives attackers time to study our networks and vulnerabilities before attacking. Additionally, once they acquire a privilege, they can maintain it for a long time. A promising approach that eliminates this situation is called the *moving target defense* (MTD), which involves changing various aspects of the network over time to shift the network's attack surface.

While several research efforts have focused on developing MTD techniques (e.g, [1]) and network level systems (e.g., [3, 5]), very little work has actually looked at characterizing the potential effectiveness of MTD systems. In this paper, we present an initial model that attempts to do just that: characterize the effectiveness of MTD systems in an enterprise network based on several system and attacker parameters.

For example, consider the scenario that can be represented by the graph shown in Figure 1. As shown, there are eight nodes in the network (i is external) with limited accessibility as shown by the edges. Thus, for attackers to compromise a specific node, say node h, they must follow the appropriate path, i.e., $i \rightarrow a \rightarrow c \rightarrow f \rightarrow h$, where the probability that attackers can compromise a node from a previous node is given on the edges, e.g. p_{ia}.

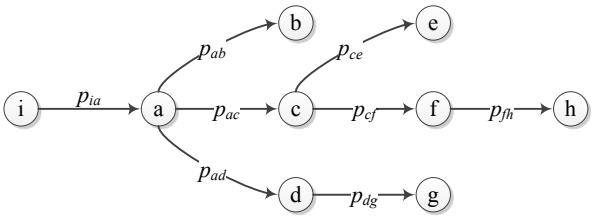

Figure 1: Example Network Graph

The goal of our model is not to compute the absolute probability of compromising a specific node, but rather to create a model that can easily compute a variety of such probabilities based on several parameters. This way, system designers can compare the use of different design parameters to compare the results in order to gain insight into how best to protect their critical nodes with an MTD system.

2. ANALYTICAL MODEL

To be useful, our model must be computationally efficient, simple, and scalable, all while clearly demonstrating the relationships between the key system parameters. Our goal is to enable a deeper understanding about how the key MTD parameters impact the security provided, which can be used to guide MTD implementation and deployment.

A key challenge in developing our MTD analytical model lies in the non-monotonic nature of MTD systems. The typical assumption that an attacker can take time to discover, compromise, and exploit an enterprise network is no longer valid within MTD systems. Using techniques such as virtual machine replacement, the attacker may lose gained privileges at any time during an intrusion attempt as the MTD system proactively adapts the network. Thus, when attacking an MTD system, the attacker must remain active to even remain in the system. Thus, we model *diligent attackers* that work inexhaustibly until they either compromise the target node or are totally removed from the system.

We first attempted to model the effect of an MTD using Markov Chains; however, as the network grew larger the state space exploded. In addition, the non-monotonicity of MTD systems breaks the Markov Chain assumption that state X_i only depends on X_{i-1} and is independent previous states. Next, we modeled nodes as states, compromises

CSIR '14 Apr 08-10 2014, Oak Ridge, TN, USA
ACM 978-1-4503-2812-8/14/04.
http://dx.doi.org/10.1145/2602087.2602088.

as transitions to new states, adaptations as transitions to previous states. Again, we ran into modeling problems as the network size increased since an attacker can actually be pushed back to any previously obtained node. From these attempts we decided to avoid backward transitions and focus only on forward and self-transitions, which represent compromising the next adjacent node and the attacker staying at current node. We capture backward transitions indirectly.

2.1 Applicability

Our model presented is suited for non-cyclic graphs such as Figure 1, although we discuss an extension in Section 4.1. However, the model is easily scalable. We also assume that communication between nodes is limited to specific paths. As discussed in [5], network-based MTD systems must ensure that system services can locate each other given the constant adaptations. This requirement, coupled with virtual machines and software controlled switches, enables the MTD to easily restrict the communications to valid paths.

As discussed above, we assume attackers have a specific target (node) in the network, e.g. a database, and must compromise the target in order to exploit it for their purposes. While attackers are limited to valid communications paths (or otherwise face easy detection, see [4]), we do assume they know those paths and continue diligently to compromise the next node in the path.

We also assume that the adaptations available to the MTD system work on particular nodes and that the adaptations not only keep the node from being compromised, but will remove attackers that have gained privileges on that node. Real world examples of such adaptations include virtual machine refreshing or replacement.

2.2 Parameters

There are five basic inputs to our model.

- Attack Interval (T_a). The time it takes to compromise a node from an adjacent node.
- Adaptation Interval (T_r). The time interval between each system adaptation.
- Number of Nodes (n). The number of nodes in the system that can be adapted by the MTD system.
- Adaptations per Adaptation Interval (k). The number of nodes ($k \leq n$) adapted during each adaptation interval (T_r).
- Attack Success Likelihood (p_{ij}). The likelihood that node j is compromised if attacked from node i in a static system. For example, in Figure 1, p_{ab} represents the likelihood of compromising node b from the node a assuming neither node is adapted.

The output of our analytical model is P_x, or the likelihood of intrusion success from outside the system (node i) to a specific node x. For example, in Figure 1, P_g represents the likelihood of intrusion success from node i to node g.

2.3 Model

To motivate our model, we use the example given in Figure 1, where the attacker tries to compromise node c by going through node a. (The same analysis can be applied to nodes b and d as well). The key concepts are illustrated in Figure 2. The values p_1, p_2, and p_3 represent the transition probabilities from i to a, a to a, and a to c that *include*

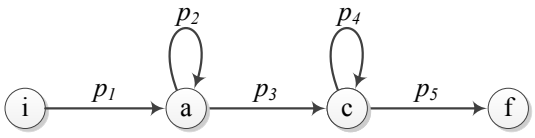

Figure 2: Transition Model for $i \rightarrow a \rightarrow c \rightarrow f$

the possibility of MTD adaptation. These probabilities are based on p_{ia}, p_{aa}, and p_{ac}, which assume *no* adaptations.

The transition probability p_1 from node i to node a is derived as follows. The probability that node a gets adapted during any adaptation interval T_a is $\frac{k}{n}$, thus the probability that a is not adapted during an adaptation is $1 - \frac{k}{n}$. Also during any attack interval T_a, there are $\frac{T_a}{T_r}$ adaptations occur. Since we assume all adaptations are independent of previous adaptations, the probability that node a does not get adapted during T_a is $(1 - \frac{k}{n})^{\frac{T_a}{T_r}}$. Putting these all together, the likelihood of a successful compromises of a from i is given by Equation 1.

$$p_1 = p_{ia} \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}} \tag{1}$$

Once inside the system (i.e., the attacker has compromised node a), the attacker can launch $\frac{T_r}{T_a}$ attacks during each time period T_r. Therefore, the probability that all attacks from a to c fail is $(1 - p_{ac})^{\frac{T_r}{T_a}}$ and, thus, the the probability that an attack from a to c succeeds during T_r is $1 - (1 - p_{ac})^{\frac{T_r}{T_a}}$.

An attacker may remain at a (following the transition from a to a) in one of two ways during time period T_a. First, the attacker may fail to penetrate from a to c and node a is not adapted. This results in a probability of staying at a of $(1 - p_{ac})^{\frac{T_r}{T_a}} \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}}$. Second, the attacker may successfully compromise node c, node a does not get adapted but node c does get adapted. This results in a second probability of staying at node a of $(1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}} \times (1 - (1 - \frac{k}{n})^{\frac{T_a}{T_r}})$. Summing these two probabilities, the ultimate probability of remaining at a is shown in Equation 2.

$$\begin{aligned} p_2 &= (1 - p_{ac})^{\frac{T_r}{T_a}} \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}} + \\ &\quad (1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}} \times (1 - (1 - \frac{k}{n})^{\frac{T_a}{T_r}}) \\ &= (1 - \frac{k}{n})^{\frac{T_a}{T_r}} - (1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{2T_a}{T_r}} \end{aligned} \tag{2}$$

Similarly, the attacker can successfully compromise c from a when both a and c do not get adapted and the attacker can follow the edge from a to c with probability $1 - (1 - p_{ac})^{\frac{T_r}{T_a}}$. Combined, this gives us the probability p_3 as shown in Equation 3.

$$p_3 = (1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{2T_a}{T_r}} \tag{3}$$

Once we know p_1, p_2, p_3, we can compute the probability of an intrusion from i to c being successful as shown in Equation 4, which relies on the simplification that since $0 < p_2 < 1$, then $p_2^0 + p_2^1 + p_2^2 + \ldots + p_2^\infty = \frac{1}{1-p_2}$.

$$P_c = p_1 \times [p_2^0 + p_2^1 + \ldots + p_2^\infty] \times p_3$$
$$= p_{ia} \times (1 - \frac{k}{n})^{\frac{T_a}{T_r}} \times \frac{1}{1-p_2} \times$$
$$(1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{2T_a}{T_r}}$$
$$= \frac{1}{1-p_2} \times p_{ia} \times (1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{3T_a}{T_r}} \quad (4)$$

Next, we consider node f in Figure 1 as the target of the attacker. (The following analysis can also similarly apply to nodes f and g.) Note that the analysis for p_1, p_2, and p_3 remains the same. Therefore, we focus on the probabilities p_4 and p_5, which represents the transition probability from $c \rightarrow c$ and $c \rightarrow f$. These probabilities can be calculated using the same analysis as for p_1, p_2, and p_3 above.

$$p_4 = (1 - \frac{k}{n})^{\frac{T_a}{T_r}} - (1 - (1 - p_{cf})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{2T_a}{T_r}} \quad (5)$$

$$p_5 = (1 - (1 - p_{cf})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{2T_a}{T_r}} \quad (6)$$

Equation 7 gives the final probability of a successful intrusion from node i to node f. This same process can also be used to compute the likelihood of a successful intrusion from i to e and g as well.

$$P_f = \frac{1}{1-p_2} \times \frac{1}{1-p_4} \times p_{ia} \times (1 - (1 - p_{ac})^{\frac{T_r}{T_a}}) \times$$
$$(1 - (1 - p_{cf})^{\frac{T_r}{T_a}}) \times (1 - \frac{k}{n})^{\frac{5T_a}{T_r}} \quad (7)$$

These derivations show that the probability of a successful intrusion to any target node is the summation of the probabilities of all possible paths to the target, which illustrates a key tenet of our analytical model. As long as the attacker can stay on a compromised node, new attacks can be launched until the target has been compromised.

2.4 General Form

To simplify applying our model to a variety of networks, we have created a general representation of the transition probabilities for any node plus the overall intrusion success probability for any target. Equation 8 captures the general form for forward transitions from node x to node y, Equation 9 captures the general form for self-transitions, and Equation 10 captures the general form for the intrusion success likelihood from i to target t. In these equations $i \rightarrow a \rightarrow b \rightarrow \ldots \rightarrow t$ represents the path from i to t, where V_p represents the nodes in the path and E_p represents the edges in this path. Here p'_{xy} and p'_{xx} represent the transition probabilities that include adaptation (e.g., $p_1 \ldots p_5$ in Equations 1 - 6) and we use T_a^{xy} instead of T_a to remove the implicit assumption that all values of T_a are identical.

3. VALIDATION

To verify our analytical model, we performed an evaluation of the intrusion success likelihood for various targets against results from the MTD simulator described in [4]. We used the network shown in Figure 1 and selected the targets of c, f, h. We ran experiments for each possible target with a fixed attack interval T_a of 100 and varied the adaptation interval T_r (20, 30, 50, 70, 80, 90, 100, 200, 300, 400, 500, 600, 700, 800, and 1,200,000 - which represents a static network). We randomly selected 1 (k) of the 8 (n) node to adapt during each adaptation interval and each static transition probability was set at 0.6. For each T_r value, we ran 20,000 intrusions against one of the specified target (c, f and h). An intrusion continued until the target was compromised or the adaptations completely removed the attacker from the system back to node i. We computed the percent of successful intrusions and compared it to the values from our analytical model.

Table 1: Model versus Experiment

	c	f	h
max deviation	0.039401	0.025922	0.027851
std deviation	0.023689	0.013295	0.012639

Figure 3 compares the calculations from our model (using Equation 10) and the experimental results for node c, f and h respectively. The first two bars in each adaptation interval represent the experimental and model values for c, the second two bars represent values for f, and the last two bars represent values for h. As expected, the likelihood of intrusion success increases as the adaptation interval increases for both the model and the experiments.

Table 1 shows the comparison between curves produced by our model against the experimental values for target nodes c, f and h. Here, the accuracy of the model is measured as the deviation between the model and the mean of the experimental results for each value of T_r. As we see, the maximum deviation for c is approximately 3.9% while the standard deviation is around 2%. These difference is even smaller for f and h, which have maximum deviation around 2.4% and standard deviation around 1.2%. We believe this accuracy is acceptable for our goal of providing insight to system designers and will support quantitative evaluation of alternative designs.

The results also clearly show the effect of path length on the intrusion success. For example, when the adaptation interval is 70, we see the effect clearly as P_c is around 30%, P_f is around 20%, and P_h is around 13%. This trend is evident throughout, although the differences between various node length decreases as the adaptation interval increases.

4. DISCUSSION AND FUTURE WORK

Our objective is not to predict the exact probabilities of an MTD system, but to provide insight to designers that allows them to make better design decisions when designing their enterprise networks. For instance, if critical data is stored at node b, a designer might want to know how to improve the security provided by the MTD system for that particular node. If we are assuming the attack interval T_a is 100, then we can pick a reasonable adaptation interval T_r and the number of adaptations per adaptation interval k, consistent with our performance requirements, to provide the required security. Another alternative would be to insert a node between a and b to increase the security. Our model will allow designers the luxury of analyzing that alternative without having to resort to expensive trial and error or simulations.

The three key parameters in our model that lie within the control of the network/MTD designer include the network node configuration (e.g., Figure 1) along with adaptation interval T_r and number of nodes adapted in each interval, n. Designers can use these along with a characterization of

$$p'_{xy} = \begin{cases} p_{xy} \times (1 - \frac{k}{n})^{\frac{T_a^{xy}}{T_r}} & \text{if } x = i; x, y \in V_p; x \to y \in E_p \\ (1 - (1 - p_{xy})^{\frac{T_r}{T_a^{xy}}}) \times (1 - \frac{k}{n})^{\frac{2T_a^{xy}}{T_r}} & \text{if } x \neq i; x, y \in V_p; x \to y \in E_p \end{cases} \quad (8)$$

$$p'_{xx} = (1 - \frac{k}{n})^{\frac{T_a^{xy}}{T_r}} - (1 - (1 - p_{xy})^{\frac{T_r}{T_a^{xy}}}) \times (1 - \frac{k}{n})^{\frac{2T_a^{xy}}{T_r}} \text{ if } x \neq i; x \to y \in E_p \quad (9)$$

$$P_t = \prod_{\substack{x \in V_p \\ x \neq i,t}} \frac{1}{1 - p'_{xx}} \times \prod_{x \to y \in E_p} p'_{xy} \quad (10)$$

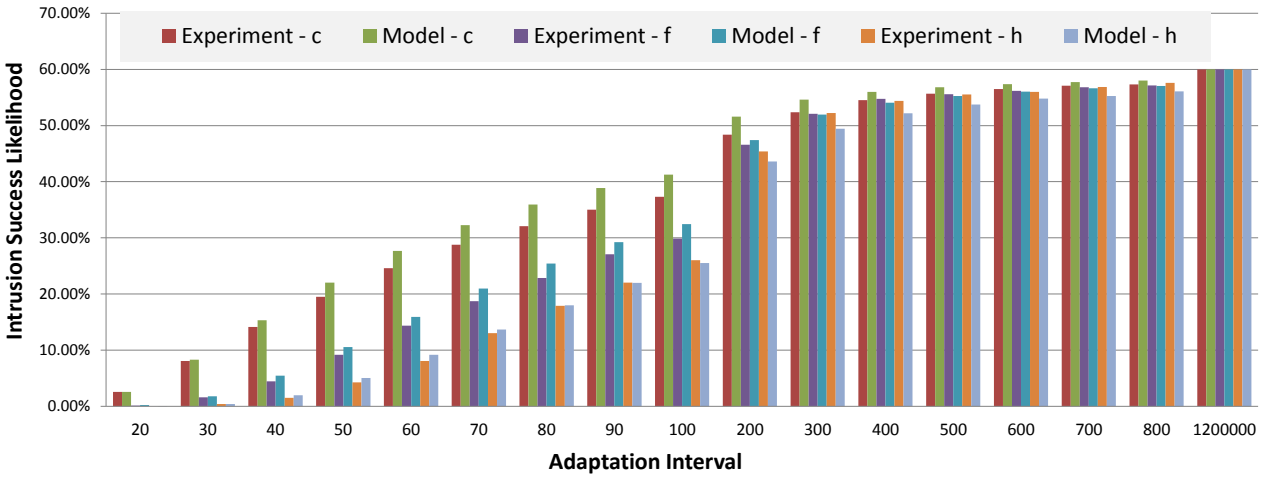

Figure 3: Model vs Experiment – Target c, f, and h

expected attack types to help tune their system.

Notice that the model is useful without knowing the exact values of the attack success likelihood between nodes. While values are required to compute the values of P_x for each x, the model can be computed against a variety of values to reveal useful insights. Likewise different values of attack intervals, T_a, may be used as well. However, there are data sets that record the time required to exploit different vulnerabilities, which can help with characterizing T_a. In addition, research on time-to-compromise models also exists [2].

4.1 Future Work

This paper represents our first foray into modeling the effectiveness of MTD systems in an enterprise network setting. As such, we have made many simplifications and assumptions. One such assumption is that the network node configuration does not contain loops. A potential solution is to consider all unique paths to the target and then sum their probabilities. Additionally, we plan to extend the model to incorporate different attack types and adaptations. One area that will require additional work is modeling how different classes of attacks affect model parameters such as T_a and how they are affected by different adaptation types. For instance, changing the IP address of a node may inhibit certain types of attacks, but it will not affect an attacker who has already compromised that node as would the node's virtual machine being refreshed or replaced. In addition, we are planning on validating the model against a real-world implementation of a MTD currently under development.

Acknowledgments

This work was supported by the Air Force Office of Scientific Research award FA9550-12-1-0106 and U.S. National Science Foundation awards 1038366 and 1018703.

5. REFERENCES

[1] S. Antonatos, P. Akritidis, E. P. Markatos, and K. G. Anagnostakis. Defending against hitlist worms using network address space randomization. *Computer Networks*, 51(12):3471–3490, 2007.

[2] M. A. McQueen, W. F. Boyer, M. A. Flynn, and G. A. Beitel. Time-to-compromise model for cyber risk reduction estimation. In *Quality of Protection*, pages 49–64. Springer, 2006.

[3] H. Okhravi, E. I. Robinson, S. Yannalfo, P. W. Michaleas, J. Haines, and A. Comella. Talent: Dynamic platform heterogeneity for cyber survivability of mission critical applications. In *Secure and Resilient Cyber Architecture Conference (SRCA'10)*, 2010.

[4] R. Zhuang, S. Zhang, A. Bardas, S. A. DeLoach, X. Ou, and A. Singhal. Investigating the application of moving target defenses to network security. In *Resilient Control Systems (ISRCS), 2013 6th International Symposium on*, pages 162–169. IEEE, 2013.

[5] R. Zhuang, S. Zhang, S. A. DeLoach, X. Ou, and A. Singhal. Simulation-based approaches to studying effectiveness of moving-target network defense. In *National Symposium on Moving Target Research*, 2012.