# Assessing the Likelihood of Cyber Network Infiltration Using Rare-Event Simulation

Alexander L. Krall, Michael E. Kuhl
Dept. of Industrial and Systems Engineering
Rochester Institute of Technology
Rochester, NY, USA, 14623
alk2345@rit.edu, Michael.Kuhl@rit.edu

Stephen F. Moskal, Shanchieh J. Yang
Dept. of Computer Engineering
Rochester Institute of Technology
Rochester, NY, USA, 14623
sfm5015@rit.edu, sjyeec@rit.edu

***Abstract*— Network infiltration is one of many types of cyber-based attacks that may be of interest to a cyber security analyst. Sufficient observation of particular events that may be uncommon during network infiltration requires special simulation techniques. This paper presents an application of the importance sampling method to estimate the likelihood of a successful network infiltration, given that sufficiently many network alerts have not been generated to achieve said success. The benefits of utilizing importance sampling within this context are assessed against the use of standard simulation.**

***Keywords—importance sampling; rare-event; alerts; network***

## I. INTRODUCTION

Network infiltration is one of many types of cyber-based attacks that may be of interest to cyber security professionals. According to Yang, Du, Holsopple, and Sudit [10], cyber-attacks have moved into a more advanced era where both security analysts and attackers are in a battle of wits, with one trying to gain the upper hand over the other with complex strategies. The result is that cyber-attacks may often take a long period before the attacker is either successful or is caught. As such, a security professional may want to assess the likelihood that certain attack actions are performed. For example, an analyst may want to see how likely an attack successfully compromises one or more machines of interest before a certain threshold number of network alerts has been reached. Said likelihood could be the determining factor in deciding whether or not network reconfiguration, by means of removing or repositioning connections, is necessary.

Noel, Jajodia, Wang, and Singhal [7] state that the removal of attack paths will inevitably reduce the options an attacker has when infiltrating a network. However, any reconfiguration to a network should be done with as much supporting data as possible. Work done by Dinh and Thai [4] provides an optimization model that determines a network configuration that minimizes the expected pairwise connectivity (number of connected node pairs) in order to build networks with greater reliability. Despite the availability of this model, determining the likelihood of infiltration within the given context can be utilized to assess whether or not any alterations have actually delivered an improvement.

This effort is supported in part by the National Security Agency under grant number H98230-15-1-0277 and the National Science Foundation under grant number 1526383.

An instance of a successful infiltration within an alert threshold may be an uncommon event by virtue of a dense or multi-leveled network configuration. In these cases, estimating the likelihood of these occurrences through simulation may require numerous trials to garner sufficient data. To achieve better results with fewer trials, we will look to employ rare-event simulation methods.

In particular, this paper will utilize importance-sampling techniques to ascertain better analysis of a rare event within the cyber network infiltration context. The rare-event of interest describes the case that a particular mission goal has been reached by an attacker, given that sufficiently many network alerts have not been generated. Section II of this paper will go on to explore various rare-event simulation method alternatives in greater detail.

## II. RELATED WORK

Two main approaches are taken towards rare-event simulation, one being splitting and the other being importance sampling. However, as of the current date, there are not many applications of rare-event simulation methods in the content of cyber security. According to Fischer, Masi, Shortle, and Chen [5], there has traditionally been a focus on detection and prevention of cyber attacks. The result was that a lesser priority was placed on modeling the impact of cyber attacks on networks. Despite these past trends in the cyber security context, analysis of the different rare-event simulation techniques can give insight as to which method is best for a particular attack type.

The rare event of interest, where an attacker reaches a particular goal given that a certain path is taken during infiltration, does not seem to be very compatible with the splitting technique. The idea of splitting incorporates the idea of starting the simulation at different states that are “close” to a rare event. Splitting is considered to be particularly useful when systems take many incremental steps on the path to a rare event [2]. Often this idea of splitting is performed in conjunction with optimization, where the intent is to reduce the variance of the final estimators. Masi, Fischer, Shortle, and Chen [6] were able to utilize splitting when modeling worm attacks on a host computer network. Computers on the network can be susceptible to the worm, infectious to other computer, or removed from the net-work due to repairs. In this case, the

rare event is defined to be that a given percentage of the susceptible machines become infected, which forms a very distinct notion of a measure of closeness. However, the concept of network infiltration has no such notion, especially when considering networks that are highly interconnected.

An alternative rare-event simulation option available is importance sampling. As opposed to splitting, importance sampling will increase the likelihood of certain actions in a network occurring by per-forming changes of measure on certain network parameters. The result is that a greater number of rare events are generated. Any data collected after implementing importance-sampling measures is then translated back in-to the framework of the original network so said data can be useful [8]. It is often noted that the "best" change of measure may be difficult to come by. One particular solution to this issue comes in the form of the Cross-Entropy (CE) method. Essentially, the CE method is an iterative process that utilizes the idea of Kullback-Leibler divergence to generate optimal changes of measure for importance sampling. CE works especially well for the exponential family of distributions, but does not work in all cases [3]. Distributions such as the continuous uniform distribution are not compatible with the method. Another issue with CE is that its algorithm seems to rely heavily on continuous data that can be ordered for the purpose of iteratively updating network parameters. Due to the requirements of the CE method, it may be difficult to utilize it in conjunction with importance sampling for the cyber infiltration context specified.

The paper will employ importance sampling within a cyber network infiltration context to assess its performance. Section III will give further details as to the nature of the problem being addressed, as well as the network configuration that will be utilized for experimentation.

## III. Rare-Event Simulation Methodology For Cyber Networks

### A. Example Cyber Network

We model an attacker's infiltration into a cyber network. The attacker has a given starting location and a given end goal. Fig. 1 depicts the network that the attacker has to progress through. Each node can be assumed to be a machine within a cyber network, where each arc is a connection from one machine to the next. Let us define $\boldsymbol{X}$ to be the set of all arcs within the network. Let $\boldsymbol{P}$ be a subset of $\boldsymbol{X}$ that represents the set of arcs along a particular path taken by the attacker. This path $\boldsymbol{P}$ will be dependent on the attacker choosing a particular direction to go in the network, should there be opportunity to do so. An attacker must compromise the target machine by exploiting any present vulnerabilities. According to Cheng, Wang, Jajodia, and Singhal [1], the common vulnerability scoring system (CVSS) is a commonly used standard to evaluate the severity of a network risk, whose scores can be converted into real probabilities for use in simulation. A particular vulnerability may be present within multiple machines within a network. Within the context of this example, the two MySQL servers (nodes 4 and 5) share the same vulnerabilities as each other. Additionally, the two backup servers (nodes 8 and 9) share the same set of vulnerabilities. All arcs are multidirectional within this network. However, once a machine has been compromised by the attacker, it remains accessible during the attack. Progression along an arc has a given likelihood of success attached to it, $\boldsymbol{p}(x)$, where $\boldsymbol{x}$ is a particular arc in the network. When a network attack begins, each attack will start at the internet (node 0) with the intent to attempt to progress to one of the two available backup servers. It is assumed that these backup servers contain some data of interest to the attacker. In all cases where there is opportunity for multiple arcs, there is a probability of choosing a particular arc when at the current node. The probability of

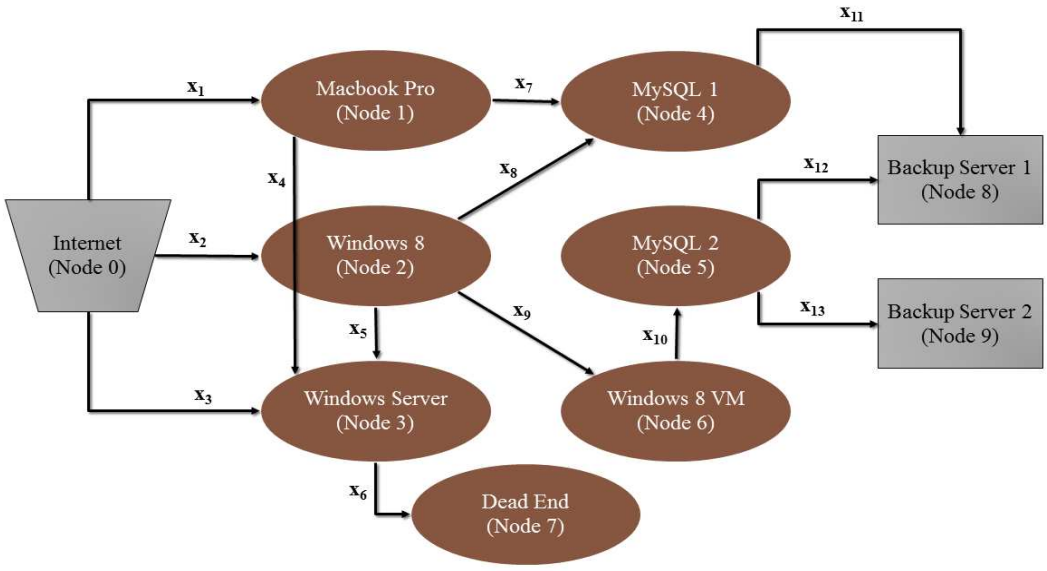



Fig. 1. Network Topography

choosing a given arc when at a particular node is given by $q(x)$. Should one arc be chosen and an attack along the chosen arc is successful, it will be included within $P$; if the attack is unsuccessful, the chosen arc will be excluded from $P$.

Another feature of cyber networks that is important to address is the idea of sensors that may be placed along arcs. Each time an attacker progresses along an arc that has a sensor attached to it, the sensor generates alerts. These alerts, reported by the sensors, typically contain various attributes with them such as the type of event [9]. A sufficient number of alerts will notify a security professional to the presence of the attacker. Within the context of this problem, it will be assumed that sensors are placed along all arcs. Should the attacker exceed some threshold number of alerts, $\delta$, the attacker will be caught and the attack will end. The addition of sensors to the model serves to add realistic factors to the cyber network. Furthermore, their addition will inevitably affect the occurrence rates of the rare event, as now the length of the path will be more important to reaching the target successfully.

Within the context of this problem, consider a set of possible end states $E$. Each end state describes a state within the simulation where the attacker cannot perform any more actions, thus the simulation ends. The following possible states for the network are elements of $E$:

- Failure state $F$, which occurs when the attacker is caught by generating too many alerts, $a$, where $a \geq \delta$.
- Goal state $G$, which will occur when the attacker reaches either machine 8 or 9, such that $a < \delta$.
- Target One State $T_{one}$, which will occur when the attacker reaches backup server one, where $T_{one} \subset G$.
- Target Two state $T_{two}$, which will occur when the attacker reaches backup server two, where $T_{two} \subset G$.

Given this information regarding alerts, we can begin to define additional attacker behavior. An attacker will continuously try to attack a chosen arc until either obtaining a success or reaching $F$. The number of attempts along a given arc is given by $k(x)$. Furthermore, an attacker will continuously try to progress further into the network. As the attacker goes through the network, it stores knowledge regarding the nodes it has compromised as well as any outgoing connections. Let the arc knowledge of the attacker at be represented by $h(i)$, where $i$ represents an event instance during the simulation. Therefore, $h(0)$ corresponds to the network upon the simulation's initialization. If reaching the dead end, the attacker will step backwards, creating groups of nodes, in order to find an outgoing connection that it has not yet compromised. Note that multiple steps backward are possible within this type of behavior and it is possible to step back to the internet node, which serves as the origin point.

The addition of this behavior is only intended to serve as a basic method in handling cases of dead ends within a network. Once detecting a point where there are uncompromised machines that can be accessed via outgoing vulnerability arcs, the attacker will then randomly choose an arc until it chooses an arc that leads to an uncompromised machine as its target. Choosing an uncompromised machine is based on the attacker knowledge at a particular event instance within the simulation. Therefore, the probability of choosing an outgoing arc must be updated to $q(x \mid h(i))$ to reflect the attacker behavior. Since the probability of success along an arc is determined by a success-until-failure scenario, the actual probability for each follows a geometric distribution, utilizing the aforementioned parameters. As such, let the probability of obtaining a particular path $P$ be given by the following:

$$f(x) = \prod_{x \in P} q(x|h_i)p(x)\big(1 - p(x)\big)^{k(x)-1} \quad (1).$$

We will define a rare event to be the instance that an attacker reaches $G$ along $P$. The variable $I_G$ takes a value of one when the rare event occurs and a value of zero otherwise.

### *B. Importance Sampling Methodology*

We will employ importance-sampling measures to the network with the intent to generate better estimates for the likelihood $\ell$ of the rare event with less work needed. One method that can be utilized to measure $\ell$ is to use Crude Monte Carlo simulation [3]. We will draw a random path sample of size $N$, where $P_1, \ldots, P_N$ are obtained from the distributions of both choosing a particular arc and of being successful along said chosen arc. Thus, we can say the following:

$$\ell = \mathbb{E}_f I_G = \frac{1}{N}\sum_{i=1}^{N} I_G \quad (2).$$

However, should the probability of reaching one of the goal nodes be sufficiently small, a larger value for $N$ will be required to yield an accurate value for $\ell$. Let $g$ be another probability density function that is continuous with respect to $f$. Utilizing $g$, we can obtain the following:

$$\ell = \int I_G \frac{f(x)}{g(x)} g(x)dx = \mathbb{E}_g I_G \frac{f(x)}{g(x)} \quad (3).$$

Note that the term $f(x)/\ g(x)$ can be defined by the likelihood ratio $L(x)$. Let $p'(x)$ and $q'(x \mid h(i))$ be changes of measure composing $g$, conducted on $p(x)$ and $q(x \mid h(i))$ respectively. Given this information, we can formally state the following regarding $L(x)$:

$$g(x) = \prod_{x \in P} q'(x|h_i)p'(x)\big(1 - p'(x)\big)^{k(x)-1} \quad (4).$$

$$L(x) = \frac{f(x)}{g(x)} \quad (5).$$

Paths $P_1, \ldots, P_N$ is a random sample from $g$. That is to say, that $P_1, \ldots, P_N$ are independently and identically distributed random vectors with density $g$. For example, should there be no change of measure in the case that $g = f$, then $L = 1$ [3]. Given that $f$ is made up by $p(x)$ and $q(x \mid h(i))$, the change of measure $g$ could incorporate changes made to either parameter (or both). Therefore, an unbiased estimator of $\ell$ is the following:

$$\hat{\ell} = \frac{1}{N}\sum_{i=1}^{N} I_G L(P_i) \quad (6).$$

From this likelihood estimator, $\hat{\ell}$, we can construct a $(100 - \alpha)$ % confidence interval. Since the idea behind importance sampling is to yield better result utilizing fewer trials, the simulation will be done in such a way to assess the amount of work needed to ascertain a confidence interval that is within some percentage $\beta$ of the likelihood. Once the confidence interval converges appropriately, the simulation does not run any more trials. In this regard, different changes of measure $g$ can be assessed against each other by comparing their final values of $N$.

## IV. Experimentation

This section will describe the specific details of the simulation study. Various cases are compared against a default base case. The base case represents the network that has not been given a change of measure and therefore represents an approach utilizing standard simulation. All modified networks are given a change of measure on at least one of their network parameters. The simulations are performed utilizing Java. Code is used from the Apache Commons Mathematics Library to calculate Z-statistics for constructing the confidence intervals.

Recall that a rare event corresponds to an attacker reaching a goal state $G$, which occurs when goal node has been reached, given that $a < \delta$. Since the simulation will run until convergence of the $(100 - \alpha)$ % confidence interval is within some percentage $\beta$ of the mean, it would potentially run for a sufficiently large time. As such, a maximum number of trials, $N_{MAX}$, is established to stop the simulation prematurely before convergence is reached. Additionally, a value for $N_{MIN}$ is set to foster meaningful convergence. Note that a simulation trial within a given run will stop when reaching any end state within the set $E$ previously established. The following parameters are used within the base case of the experiment:

- $M = 60$
- $N_{MAX} = 1.0 \cdot 10^7$
- $N_{MIN} = 1.0 \cdot 10^3$
- $\alpha = 0.05$
- $\beta = 0.01$
- $\delta = 10$
- $q(x_1 \mid h(0)) = q(x_2 \mid h(0)) = q(x_3 \mid h(0)) = q(x_5 \mid h(0)) = q(x_8 \mid h(0)) = q(x_9 \mid h(0)) = 0.33$
- $q(x_4 \mid h(0)) = q(x_7 \mid h(0)) = q(x_{12} \mid h(0)) = q(x_{13} \mid h(0)) = 0.50$
- $q(x_6 \mid h(0)) = q(x_{10} \mid h(0)) = q(x_{11} \mid h(0)) = 1.00$
- $p(x_1) = 0.88$
- $p(x_2) = 0.73$
- $p(x_3) = p(x_4) = p(x_5) = 0.43$
- $p(x_6) = 0.80$
- $p(x_7) = p(x_8) = p(x_{10}) = 0.40$
- $p(x_9) = 0.65$
- $p(x_{11}) = p(x_{12}) = p(x_{13}) = 0.01$

Due to the low probability of success for compromising either of the two backup servers, importance sampling is performed on the $p(x_{11})$, $p(x_{12})$ and $p(x_{13})$ parameters for the modified networks. The change of measure should be performed on these parameters since the backup servers share the same vulnerabilities, as has been previously indicated. For the experiment, $p(x_{11})$, $p(x_{12})$ and $p(x_{13})$ will be set equal to $p(\lambda)$.

### A. Experimentation Results

Simulation results for each of the cases tested during experimentation are shown in Tables I – III. In particular, Table I displays the data pertinent to all rare events. Table II only contains information regarding compromising backup server one and Table III only contains information regarding compromising backup server two. The measures of interest consist of the average number of trials before convergence, average likelihood (of a rare event occurring), likelihood variance, and average confidence interval half width.

TABLE I. Simulation estimates for each case tested

| Case $p(\lambda)$ | Type | Average Required Trials | Average Likelihood | Likelihood Variance | Average Half-Width |
|---|---|---|---|---|---|
| 0.01 | Base | 9.254E+05 | 3.986E-02 | 4.144E-08 | 3.986E-04 |
| 0.02 | Modified | 4.549E+05 | 3.982E-02 | 3.295E-08 | 3.982E-04 |
| 0.03 | Modified | 2.979E+05 | 3.983E-02 | 2.425E-08 | 3.983E-04 |
| 0.04 | Modified | 2.193E+05 | 3.986E-02 | 4.223E-08 | 3.986E-04 |
| 0.05 | Modified | 1.726E+05 | 3.982E-02 | 3.922E-08 | 3.982E-04 |
| 0.06 | Modified | 1.412E+05 | 3.985E-02 | 2.240E-08 | 3.985E-04 |
| 0.07 | Modified | 1.191E+05 | 3.981E-02 | 3.394E-08 | 3.981E-04 |
| 0.08 | Modified | 1.023E+05 | 3.985E-02 | 5.404E-08 | 3.985E-04 |
| 0.09 | Modified | 8.944E+04 | 3.983E-02 | 4.682E-08 | 3.983E-04 |
| 0.10 | Modified | 7.915E+04 | 3.984E-02 | 3.117E-08 | 3.984E-04 |

All average values represent the mean value of the relevant measure of interest across all $M$ replications for a particular case. Additionally, the likelihood variance is taken with respect to the final likelihood values for all $M$ replications for the case considered; this is done to serve as an indicator as to how consistent likelihood estimates are between replications for a particular case.

TABLE II. BACKUP SERVER 1 SPECIFIC DATA

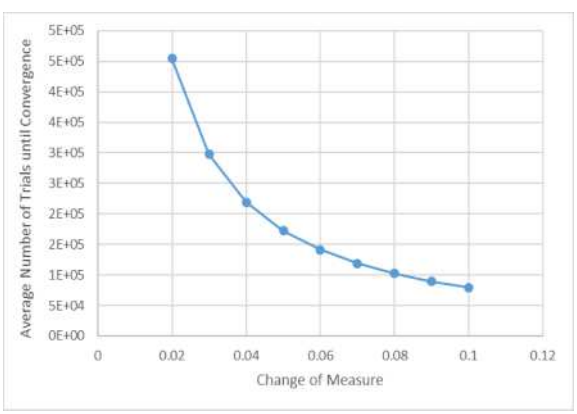
| Case $p(\lambda)$ | Type | Average Likelihood | Likelihood Variance | Average Half-Width |
|---|---|---|---|---|
| 0.01 | Base | 3.681E-02 | 3.370E-08 | 3.836E-04 |
| 0.02 | Modified | 3.678E-02 | 2.917E-08 | 3.840E-04 |
| 0.03 | Modified | 3.678E-02 | 2.500E-08 | 3.847E-04 |
| 0.04 | Modified | 3.682E-02 | 3.926E-08 | 3.857E-04 |
| 0.05 | Modified | 3.678E-02 | 3.854E-08 | 3.860E-04 |
| 0.06 | Modified | 3.681E-02 | 2.701E-08 | 3.871E-04 |
| 0.07 | Modified | 3.675E-02 | 2.954E-08 | 3.874E-04 |
| 0.08 | Modified | 3.680E-02 | 4.737E-08 | 3.887E-04 |
| 0.09 | Modified | 3.679E-02 | 3.866E-08 | 3.894E-04 |
| 0.10 | Modified | 3.679E-02 | 3.557E-08 | 3.902E-04 |

Upon analysis of the values in Table I, it can be seen that the modified cases require significantly less trials. When $p(\lambda) \geq 0.09$, the number of trials required before reaching convergence is reduced by an order of magnitude.

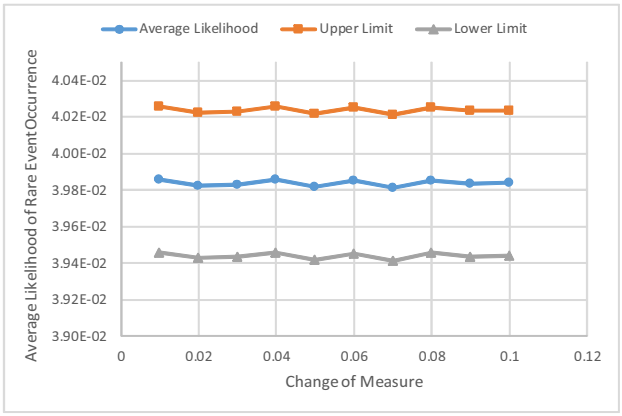
TABLE III. BACKUP SERVER 2 SPECIFIC DATA

| Case $p(\lambda)$ | Type | Average Likelihood | Likelihood Variance | Average Half-Width |
|---|---|---|---|---|
| 0.01 | Base | 3.046E-03 | 3.968E-09 | 1.123E-04 |
| 0.02 | Modified | 3.040E-03 | 2.889E-09 | 1.141E-04 |
| 0.03 | Modified | 3.048E-03 | 3.846E-09 | 1.163E-04 |
| 0.04 | Modified | 3.040E-03 | 3.768E-09 | 1.183E-04 |
| 0.05 | Modified | 3.040E-03 | 3.172E-09 | 1.203E-04 |
| 0.06 | Modified | 3.043E-03 | 4.060E-09 | 1.226E-04 |
| 0.07 | Modified | 3.063E-03 | 3.695E-09 | 1.251E-04 |
| 0.08 | Modified | 3.050E-03 | 3.658E-09 | 1.272E-04 |
| 0.09 | Modified | 3.041E-03 | 4.813E-09 | 1.291E-04 |
| 0.10 | Modified | 3.050E-03 | 3.806E-09 | 1.317E-04 |

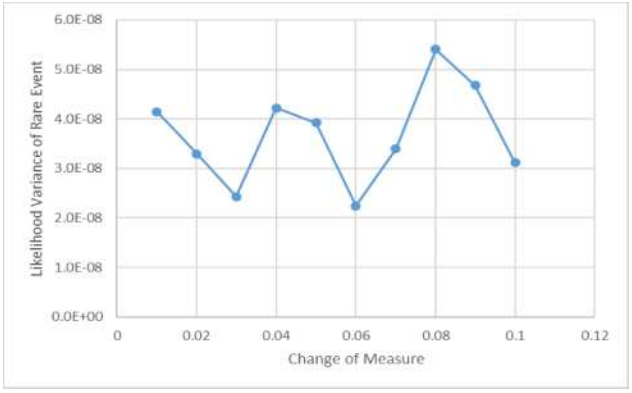
Additionally, the average likelihood and half-width values are consistent among all cases, indicating that importance




(a)

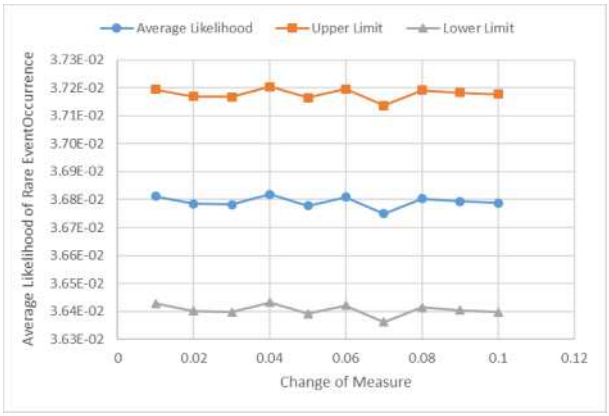



(b)




(c)

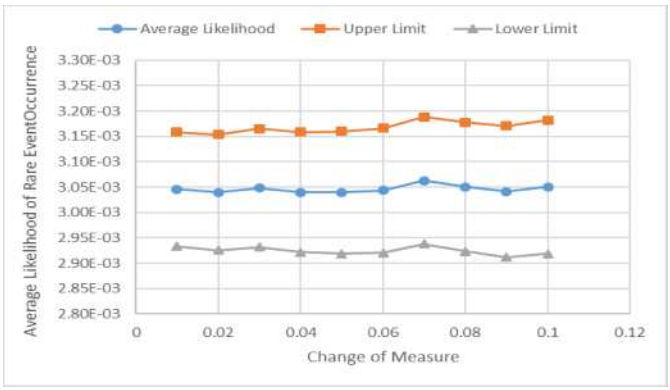
Fig. 2. Rare Event Simulation Output (All Events)



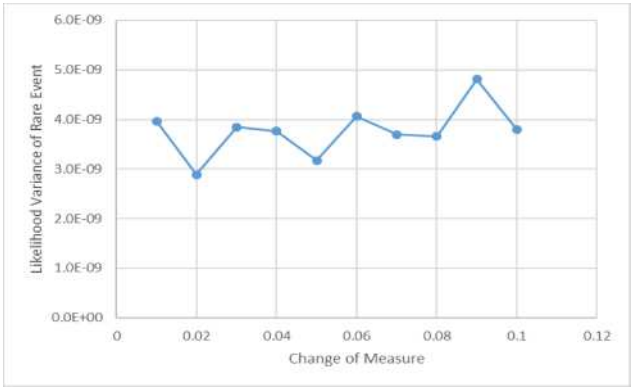

(a)


(b)

Fig. 3. Backup Server One Simulation Output




(a)




(b)

Fig. 4. Backup Server Two Simulation Output

sampling is working as intended. Furthermore, the order of magnitude of the likelihood variance lies within the same order of magnitude for all cases, which gives further confirmation that the likelihood values per replication do not differ greatly. Similar trends can be seen in Tables II – III.

Fig. 2, Fig. 3, and Fig. 4 displays a graphical representation of the data contained in Tables I – III respectively. Within this, Fig. 2 (a) displays the average number of trials for each modified case, representing $0.02 \leq p(\lambda) \leq 0.10$ The base case where $p(\lambda) = 0.01$ is not displayed on the chart because it obfuscates the relationship between $p(\lambda)$ and the average number of required trials due to its significantly large value. The overall trend seen in Fig. 2 shows that the average number of required trials is exponentially decreasing in such a way that it seems to be converging on some value. Essentially, increasing the value of $p(\lambda)$ begins to have diminishing returns. At a certain point, increasing $p(\lambda)$ provides negligible reduction in the required number of trials before convergence.

Fig. 2 (b) displays the average likelihood of the rare event occurring along with the confidence interval lower and upper limits. Note that all modified cases are shown along with the base case. The overall uniformity of the estimates shows that convergence has occurred for all cases, indicating that the importance sampling technique is yielding accurate estimates. The notion that importance sampling is working as expected is reinforced by the fact that the likelihood estimations for all modified cases are consistent with that of the base case. The legend seen in Fig. 2 (b) identifies the lower limit, likelihood and upper limit values for clarity. Similar trends can be seen in Fig. 3 (a) and Fig. 4 (a).

In regards to the likelihood variance, no discernable trend can be seen in Fig. 2 (c), Fig. 3 (b), or Fig. 4 (b). The values seem to oscillating up and down without any assignable cause. Some of the modified cases have less variable estimates for the rare-event likelihood, while others have worse.

## V. Conclusion and Future Work

Cyber network infiltration is of particular interest to security analysts. In this paper, we have applied importance-sampling techniques within different cases in order to model network infiltration attacks. The rare event that a goal node has been reached given that sufficiently many network alerts have not been generated had its likelihood estimated. Importance sampling techniques provided good estimates for the rare event likelihood while simultaneously requiring a fewer number of trials when compared to the base case.

### A. Future Work

Future work will apply this context of rare-event simulation to the concept of Moving Target Network Defense (MTD). Put simply, MTD can be interpreted "as the fact that the network is constantly changing to reduce/shift the attack surface area available for exploitation by attackers" [11]. The dynamic nature of MTD would serve to reduce the likelihood of events in a network. Application of rare-event simulation within the context of cyber network infiltration as it has been applied in this paper may be useful for modeling this form of problem. Lastly, the importance sampling technique used in this paper had changes of measures that were set manually. Looking into some form of method that automatically populates the optimal change of measure for this type of cyber security problem may also be of use.

## ACKNOWLEDGMENT

This effort is supported in part by the National Security Agency under grant number H98230-15-1-0277 and the National Science Foundation under grant number 1526383. This manuscript is submitted for publication with the understanding that the United States Government is authorized to reproduce and distribute reprints.

## REFERENCES

[1] P. Cheng, L. Wang, S. Jajodia, and A. Singhal, "Aggregating CVSS Base Scores for Semantics-Rich Network Security Metrics," *2012 IEEE 31st Symposium on Reliable Distributed Systems*, 2012.

[2] B. Crain, C.H. Chen, and J. F. Shortle, "Combining simulation allocation and optimal splitting for rare-event simulation optimization," *Proceedings of the 2011 Winter Simulation Conference (WSC)*, 2011.

[3] P.T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A Tutorial on the Cross-Entropy Method," *Annals of Operations Research Ann Oper Res*, vol. 134, no. 1, pp. 19–67, 2005.

[4] T. N. Dinh and M. T. Thai, "Assessing attack vulnerability in networks with uncertainty," 2015 *IEEE Conference on Computer Communications (INFOCOM)*, 2015.

[5] M. J. Fischer, D. M. Masi, J. F. Shortle, and C.H. Chen, "Simulating non-stationary congestion systems using splitting with applications to cyber security," *Proceedings of the 2010 Winter Simulation Conference*, 2010.

[6] D. M. Masi, M. J. Fischer, J. F. Shortle, and C.H. Chen, "Simulating network cyber attacks using splitting techniques," *Proceedings of the 2011 Winter Simulation Conference (WSC)*, 2011.

[7] S. Noel, S. Jajodia, L. Wang, A. Singhal, "Measuring Security Risk of Networks using Attack Graphs," *International Journal of Next-Generation Computing*, vol. 1, no. 1, pp. 135-147, 2010.

[8] P. Shahabuddin, "Importance Sampling for the Simulation of Highly Reliable Markovian Systems," *Management Science*, vol. 40, no. 3, pp. 333–352, 1994.

[9] L. Wang, A. Liu, and S. Jajodia, "Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts," *Computer Communications*, vol. 29, no. 15, pp. 2917–2933, 2006.

[10] S. J. Yang, H. Du, J. Holsopple, and M. Sudit, "Attack Projection," *Advances in Information Security Cyber Defense and Situational Awareness*, pp. 239–261, 2014.

[11] R. Zhuang, S. Zhang, S.A. DeLoach, X. Ou, A. Singhal, "Simulation-based Approaches to Studying Effectiveness of Moving-Target Network Defense," *National Symposium on Moving target Research*, pp. 1–12, 2012.