International Journal of Security and Its Applications
Vol.9, No.2 (2015), pp.123-134
http://dx.doi.org/10.14257/ijsia.2015.9.2.12

Analysis of Port Hopping for Proactive Cyber Defense[1]

Yue-Bin Luo, Bao-Sheng Wang, Gui-Lin Cai

College of Computer, National University of Defense Technology
410073 Changsha, China
{luoyuebin, bswang, glcai}@nudt.edu.cn

Abstract

Port hopping is a typical proactive cyber defense technology, which hides the service identity and confuses attackers during reconnaissance by constantly altering service ports. Although several kinds of port hopping mechanisms have been proposed and implemented, but it is still unknown how effective port hopping is and under what circumstances it is a viable moving target defense because the existed works are limited and they usually discuss only a few parameters. Besides, in many cases the defense effectiveness has been studied empirically.

In order to have an insight into the effectiveness of port hopping, this paper introduces a quantitative analysis based on the urn model, which quantifies the probability of attacker success in terms of port pool size, number of probes, number of vulnerable services, and hopping frequency. Theoretical analysis shows that port hopping is an effective and promising proactive defense technology in thwarting cyber attacks.

Keywords: *Port Hopping, Network Security, Moving Target Defense, Proactive Defense*

1. Introduction

The Internet is becoming pervasive today with the emergence of new technologies and devices which provide anytime anywhere network connections, which brings us enormous benefits and conveniences in an unbelievable way. Meanwhile, it also makes users connected to the Internet suffer more attacks and threats, such as eavesdropping, worms, deceptions and DoS/DDoS [1-2] attacks. Internet has become a double-edge sword. For the purpose of addressing these problems, several conventional security mechanisms such as firewall and intrusion detection/prevention system (IDS/IPS) [3-5] have been proposed. However, most of these countermeasures are insufficient to protect legitimate users from malicious attackers because they are passive and incomplete in nature. In currently deployed systems, the attacker has a static target to perform reconnaissance, find vulnerabilities, and gain privileged access on target's machine and network, while the defender has no effective count measures until the exploits noticed, vulnerability found, patch released, and then applied widely. The dynamics of this process significantly favors the attacker over the defender because the attacker exists in the dark side while the defender in the bright, the attacker needs to find only a single exploitable system bug while the defender must ensure none exist. Intelligent attackers may cause bad damages to the victims through sophisticated attacks even if conventional security defense systems are completely applied, and this is the reason why cyber attacks develop so fast.

[1] This paper is a revised and expanded version of a paper entitled "Effectiveness of Port Hopping as A Moving Target Defense" presented at SecTech 2014, Hainan China, December 20-23.

ISSN: 1738-9976 IJSIA

In order to turn the asymmetric scale of cyber attacks and defenses, people focus their attentions on moving target defense (MTD) [6-7] technologies, such as address and port hopping, which is a novel proactive cyber defense mechanism where the defenders continuously shift their systems' attack surfaces to increase attackers' efforts in performing reconnaissance and compromising their systems [8]. MTD alters systems' network attributes in an unpredictable way, and in such a way to confuse potential attackers and render their knowledge ineffective or impractical. As a result, the attacker will act on false information that can result in expending higher attack price and increasing the risk of being detected. Port hopping is a typical MTD mechanism which dynamically maps a service's port number to an unused pseudo-random one. In this way, port hopping disrupts the static associations between port numbers and services, which can be used to mitigate reconnaissance attack because the attacker often identifies services of a given system by network probes, and then breaks in the target system by exploiting a known vulnerability of the service identified.

Although several MTD mechanisms [9-10] have been proposed, this paper examines the effectiveness of port hopping [11-15]. These existed works show that port hopping is a promising technique that transforms port number used for communication with random one so as to improve the ability of thwarting cyber attacks. However, the performance evaluation has largely been empirical, and the results of these researches are typically limited to very specific scenarios. Our early work in [16] introduced urn model and proposed a novel evaluation model of effectiveness of port hopping. In this paper, we use urn models to give a more exhaustive evaluation about the effectiveness of port hopping, and quantify the likelihood of attacker success in terms of port pool size, number of probes, number of vulnerable services, and hopping frequency.

The contributions of this work are summarized as follows.

- This paper introduces probabilistic models based on urn models and provides quantitative analysis tools to evaluate the effectiveness of port hopping defense.
- This paper derive a set of probabilistic models to quantify the ASR in terms of port pool size, number of probes, number of vulnerable services, and hopping frequency.
- This paper validates the effectiveness of port hopping through simulations and experiments.

The remainder of this paper is organized as follows. Section 2 analyzes the existing implementations and evaluations of port hopping and briefly describes the motivation of this work. Section 3 introduces a probabilistic model and describes two extreme application scenarios. Section 4 uses urn model to give detailed evaluations of port hopping under different conditions. Section 5 summarizes the effectiveness of port hopping and concludes this paper.

2. Motivation and Related Works

The Server/Client model is the main communication form of computer networks, where communication entities establish communication channels for data transformation based on sockets. In the service process, the server accepts messages at a well-known port (0 through 1023). That is, each server receives its messages at a fixed port that is widely published, for example, the web server listens for requests at well-known port 80 on each host, the domain name server (DNS) receives messages at port 53, and the mail service listens for messages at port 25, and so on. The static well-known ports mechanism is convenient for clients to initiate connections; however, it also brings severe security problems since it is also convenient for attackers to perform cyber attacks such as port scanning and eavesdropping. Once an exploitable vulnerability is discovered, DoS/DDoS attacks can be launched against the target systems immediately.

The hacking processes of the current cyber attacks typically progress according to a certain attack chain that begins with system reconnaissance where the attacker seeks to gather valuable information about the target system. The attacker uses network scanners such as *Nmap*, *Amap* and *TTLscan et al.* [17] to gather the target's IP address, operating system, and service information, which is important since it can be used to determine and develop subsequent exploit mechanism. Therefore mitigating the reconnaissance phase of a cyber attack can be an effective defense strategy. Typically, in the conventional defense technologies, the firewall is used to manage ingress and egress flows of the protected network based on rules set by administrator beforehand, which proved insufficient to protect users' systems since it is passive and incomplete in nature.

Port hopping is a proactive defense mechanism which constantly maps the well-known port of the protected service to an unused pseudo port randomly chosen from the port pool available to the administrator, which provides a diversity defense where the service port changes constantly in an unpredictable way. Port hopping is an active reconnaissance defense that periodically permutes the relationships between ports and services, which introduces additional uncertainty that causes the attacker either expends additional efforts in attack preparations or assumes greater risk in the subsequent attack process.

Several port hopping implementations have been proposed before:

Michael Atighetchi *et al.* [11] develop a port and address hopping mechanism based on time synchronization and random number generation, where hopping delegate and network address translation (NAT) gateway perform mapping between fake (*address* : *port*) and real (*address* : *port*) pairs. Henry C.J. Lee and Vrizlynn L.L. Thing [12] propose a random port hopping technique, where UDP/TCP port number used by the server varies as a function of time and a shared secret between the server and the receiver client. Gal Badishi *et al.* [13-14] introduce an ACK-based method and overcome the problem of synchronization in port hopping, and carry out the simulation under the normal attack patterns to validate it. Shi Leyi *et al.* [15] introduce a timestamp-based service hopping technique, and performed theoretical analysis on hopping service. Zhang Fu *et al.* [18] also propose algorithms, called *BiGWheel* and *HoPerAA*, to communicate with multiple receivers in a port-hopping manner and to realize the RPH in the presence of clock-drift. Kousaburou Hari and Tadashi Dohi [19-20] develop a simulator to carry out the behavioral analysis of the port hopping communication model, and take place the sensitivity analysis of port hopping in terms of the communication success rate.

Although several implementations of port hopping existed, and limited testing has demonstrated that port hopping methods are effective in reconnaissance defenses, their scopes are limited and only a few parameters have been discussed. Besides, in many cases the defense performance has been studied empirically. This paper develops theoretical models based on urn models to give a better understanding of the effectiveness of port hopping under various conditions. We use Urn models to examine the attacker's attack success rate (ASR) under static ports and perfect port hopping. In a static ports network where services' ports are fixed, the optimal strategy of an attacker is to serially scan the entire port space so that all of the active service ports will be found. By contrast, in the perfect port hopping network where a hopping event happens every time a probe attempts. Even if the attacker is able to serially scan the entire port space, it is unlikely that all active service ports would be discovered. The main contributions of this work is in introducing a probabilistic model which can be used to derive a comprehensive quantitative analysis of port hopping, and validate the defense performance through simulations and experiments.

3. Analytical Models

To be useful, our model must be simple while clearly demonstrating the relationships between the key hopping parameters and defense effectiveness. Our goal is to demonstrate how the key port hopping parameters impact the security performance provided, which can be used to guide port hopping implementation and deployment.

3.1. Model Abstraction

Consider a scenario as follows: a server host needs to keep several ports open to provide standard network services; meanwhile, an attacker targets the server protected by an administrator. In the reconnaissance phase, the attacker seeks to perform reconnaissance on the host while the administrator tries to conceal the attributes of the current active services. The reconnaissance will succeed once the attacker finds an active port of a vulnerable service, which can be converted to urn models in the statistics theory.

In an urn model, objects of real interest (such as atoms, people, cars, etc.) are represented as colored balls in an urn or other container. In the basic urn model, the urn contains x white and y black balls; one ball is drawn randomly from the urn and its color observed; it is then placed back or not placed back into the urn according to different models, and the selection process is repeated. Although simple in sought, urn models are useful statistic tools to modeling various complex systems.

For a particular network server host, consider an urn model containing v black and n - v white balls for total of n balls. The population of balls represents the number of service ports available to the administrator. The black balls represent vulnerable services ports, while the white balls represent secure service ports such as a port not in use or a port which is active but not associated with a vulnerable service.

The attacker draws k balls once at a time from the urn. If the attacker draws at least one black ball that represents the vulnerable service port, then the set of draws that represents the process of attack is considered a success. Altering what happens in the intervals of different draws can be used to represent different defense strategies. Let’s consider two extreme scenarios of port hopping: static port (no hopping) and perfect port hopping (hopping after every probe).

3.2. Parameters and Assumptions

3.2.1. Parameters: In order to give a quantitative analysis about the effectiveness of port hopping, some basic parameters should be predefined.

- n: The number of ports (n < 65536) available in the system, which can be associated to network services.
- v: The number of ports ($v \leq n$) associated with active vulnerable services.
- k: The number of probes that the attacker performs in the reconnaissance phase.
- m: The number of probes allowed before one port hopping.
- x: The number of vulnerable ports ($x \leq v$) found in k probes.
- $P(x)$: The probability of x vulnerable ports found in k probes, so $P(x > 0)$ represents the ASR.

3.2.2. Assumptions: We make some assumptions regard to the attacker's hacking capabilities and the administrator's defense abilities as follows.

- The attacker knows the address of server host, and can perform reconnaissance to the target.
- The attacker is aware of the port pool (*n* ports) and will serially attempt *k* connections (probes).
- A hopping event alters all ports active in the server system to new ports selected randomly from the unused port pool.
- The goal of the attacker is to contact at least one of the *v* vulnerable ports in *k* attempts, and then the attack succeeds.

3.3. Static Ports

In the conventional system, the ports assigned to services in the server are fixed, which is known as static ports. In this situation, the simplest strategy that the attacker needs is to sequentially scan the whole ports space. If $k \geq n$, the attacker will find all *v* vulnerable services. However, if $k < n$ it is not sure whether the attacker will find vulnerable services. In this situation, we can use urn model to quantify the likelihood of attack success.

Given the setting of static ports, where the attacker doesn't need to scan the port which is canned before. This attack scenario can be quantified by the urn model of sampling without replacement, where the ball drawn from the urn will not be placed back. In this situation, if *X* is a random variable which represents the number of black balls drawn in a sequence of *k* draws. The attack success rate is determined by a hypergeometric distribution where

$$P(X = x) = \frac{\binom{v}{x}\binom{n-v}{k-x}}{\binom{n}{k}} \ . \tag{1}$$

In this case, the sampling without replacement models the information gained by the attack before is helpful to the following reconnaissance, then the attack success rate is

$$P(X \geq 1) = 1 - P(X = 0) = 1 - \frac{\binom{n-v}{k}}{\binom{n}{k}} \ . \tag{2}$$

3.4. Perfect Port Hopping

Port hopping randomly alters the port which is connected to an active service. As previously stated, we assume a hopping event alters all ports active in the server system to new ports selected randomly from the unused port pool. Therefore, the moment a hopping event happens, the attacker loses all ports probed. The hopping frequency will impact the success rate of the attack. The extreme case is perfect port hopping, where a hopping event happens every time a probe attempts. The knowledge which the attacker gained in one probe will not be useful to another, thus the attacker needs to probe ports scanned before more than one time. Even if the attacker is allowed $k = n$ attempts, it is unlikely that all active service ports would be discovered and the attack success probability is still less than one.

In the circumstance of port hopping, the attack scenario can be quantified by the urn model of sampling with replacement, in other words, the ball drawn from the urn will be place back. In this situation, if *X* is a random variable which represents the number of black balls drawn in a sequence of *k* draws. The attack success rate is determined by a binomial distribution and

$$P(X = x) = \binom{v}{x} p^x (1-p)^{k-x} \ . \tag{3}$$

where $p = \frac{v}{n}$ is the probability of drawing a black ball (vulnerable service), and the success likelihood of attack in given *k* probes is

$$P(X \geq 1) = 1 - P(X = 0) = 1 - (1-p)^k \ . \tag{4}$$

In the next section, we will use these models to analyze the effectiveness of port hopping.

4. Analysis of Hopping Defenses

The previous section introduced statistic models for two extreme cases of port hopping, static ports and perfect port hopping. The attack success rates depend on a variety of factors, such as the port pool size, number of probes, number of vulnerable services, and the hopping frequency. The influences of these factors under certain conditions will be analyzed in this section.

4.1. Port Pool Size

Consider the scenario where there are *v* = 1, 2, ⋯, 5 vulnerable services existed in the server host, and the attacker is capable of probing the entire port space (*k* = *n*). In this case, the static ports provide no defense since the attacker can scan the entire port space (0 ~ 1023, at most, 0 ~ 65535) and all of the vulnerable services will be found. By contrast, the port pool size needs to be considered in port hopping defense, where the port pool size represents the number of ports available for port hopping (0 ~ *n*, and $n \leq 65536 - 1024 = 64512$ in order to not causing a collision with the well-known port space 0 ~ 1023).

In the case of perfect port hopping, the Equation (4) becomes

$$P(0 < X \leqslant n) = 1 - (1 - \frac{v}{n})^n \ . \tag{5}$$

As shown in Figure 1, the attack success rate decreases as the port pool size increases, and the attack success rate is really high when the port pool size is small. The attack success rate is close to 0.99 when *v* = 5, and it falls as the port pool size increases and ultimately to $e^{-1} \approx 0.63$ when *k* = *n*, *v* = 1. In this case, perfect port hopping reduces the attack success rate by 37% as compared to using static ports.

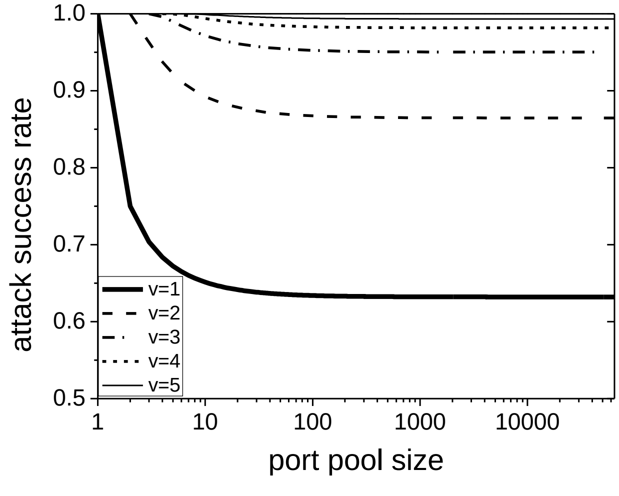

Figure 1. ASR vs the Port Pool Size

4.2. Number of Probes

The number of probes ($k \leq n$) permitted is another factor which has an effect on the attack success rate. Generally, the operating systems introduce some strategies to mitigate network attacks, so the number of probes is restricted. Again, we assume there are $v = 1, 2, \cdots, 5$ vulnerable services existed in the server host.

In this case, the attack success rate against static ports mechanism defined by Equation (2) becomes

$$P(0 < X \leqslant k) = 1 - \frac{\binom{n-v}{k}}{\binom{n}{k}} \ . \tag{6}$$

As shown in Figure 2, the attack success rate against static ports mechanism increases linearly as the number of probes increases. Especially, Equation (6) equals to k/n when $v = 1$, which is plotted in Figure 2 with thin dashed line.

For perfect port hopping, the likelihood of attack success defined by Equation (4) becomes

$$P(0 < X \leqslant k) = 1 - (1 - \frac{v}{n})^k \ . \tag{7}$$

The attack success rate in this scenario is shown in Figure 2, the likelihood of attack success increases slowly as the number of probes increases, which is slower than that of static ports when v is fixed. Figure 2 also shows that the attack success rate increases as the number of vulnerable services increases when number of probes k is fixed, and the fewer vulnerable services a server host has, the better defense performance port hopping performs. As the previous section, and the ultimate value of the attack success rate is 0.63 when $v = 1$ and the entire port pool can be probed. In comparison, the attack success rate is close to 0.99 when $v = 5$.

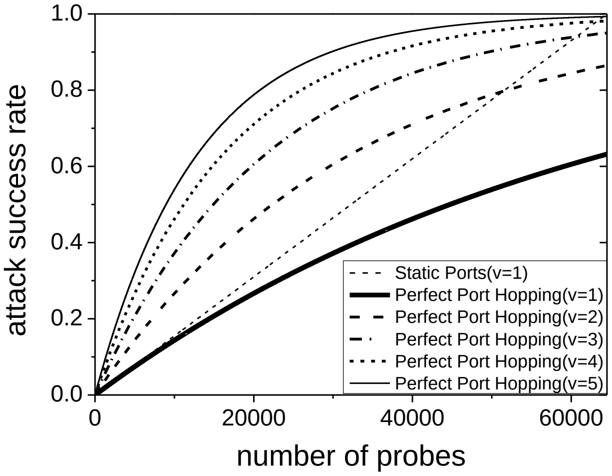

Figure 2. ASR vs the Number of Probes

Compared to static ports, perfect port hopping provides an improved defense.

4.3. Number of Vulnerable Services

Another key factor which affects the attack success rate is the number of vulnerable services existed in the server host. Compared to the previous assumption where there are $v = 1, 2, \cdots, 5$ vulnerable services, in this section, the scenario of multiple vulnerable services is considered. The attacker's goal is to discover at least one of the v vulnerable services. Assuming the attacker is permitted $k = n$ probes, the static ports mechanism provides no defense because all of v vulnerable services will be found. For perfect port hopping, assuming the maximum port pool (N = 64512) is used, then the likelihood of attack success defined by Equation (4) becomes

$$P(0 < X \leqslant k) = 1 - (1 - \frac{v}{N})^N \ . \tag{8}$$

Obviously, in Figure 3, the attack success rate increases as the number of vulnerable services increases, and the attack success rate is close to 1 when the number of vulnerable services $n \geq 5$. Therefore, port hopping is an acceptable defense if there is a small population of vulnerable services. The ultimate value of the attack success rate is 0.63 when v = 1, which is in consistent with previous sections.

4.4. Hopping Frequency

Hopping frequency is another important factor which affects the effectiveness of port hopping. In this section, the influence of hopping frequency will be studied.

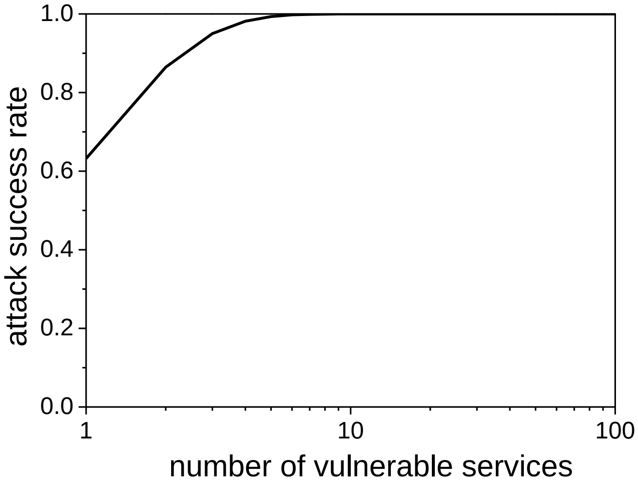

Figure 3. ASR vs the Number of Vulnerable Services

Again, in this section, we assume the attacker is permitted k = n probes and the maximum port pool (N = 64512) is used. The hopping frequency varies from no hopping (static ports) to perfect port hopping (hopping after every probe), which is normalized as [0, 1], where 0 represents no hopping (in this case, m can be seen as m = N) and 1 represents perfect port hopping (in this case, m =1). Therefore, there is N/m hopping events happened in the reconnaissance lifetime of probing the entire port pool, and the attack success likelihood defined by Equation (4) becomes

$$\begin{aligned} P(X > 0) &= 1 - P(X = 0) \\ &= 1 - P(X_1 = 0)P(X_2 = 0) \cdots P(X_{\frac{N}{m}} = 0) \\ &= 1 - [\frac{\binom{N-v}{m}}{\binom{N}{m}}]^{\frac{N}{m}} \ . \end{aligned} \tag{9}$$

As shown in Figure 4, we give the analysis results of five different scenarios where the number of the vulnerable services existed in the server host is v = 1, 2, ⋯, 5, and the likelihood of attack success decreases as the hopping frequency increases when v is fixed. In consistent with previous sections, the ultimate likelihood of attack success is approximately 63% when perfect port hopping performed in the scenario where v = 1. In comparison, the attack success rate is close to 99% when v = 5. Meanwhile, theoretical analysis in this section also shows that port hopping has a good defense performance if the number of vulnerable services existed in the system is small.

Without considering the overhead of port hopping, the perfect port hopping is the best defense mechanism we should take when the number of vulnerable services is fixed. Actually, in the existed implementations of port hopping, the overhead of port hopping grows exponentially as the hopping frequency increases. Therefore, in the implementation of a port hopping system, the system overhead caused by hopping events must be considered. According to the influences of hopping mechanisms, the implementations can be classified to three categories. The first kind of implementation, where a hopping event breaks all of the active sessions and the users need to establish connections again after port hopping happens. The second kind, where the port connected to an active session will be kept to the end of the session so a service may has more than one active port at one time, which also brings convenience to the attack at the same time. The last kind, where hopping delegate performs mapping between fake random ports and real service ports in real time, and the active connections will not be disrupted. However, no matter which category a port hopping belongs to, it causes a definite computation and communication overhead undoubtedly. Therefore, trade-offs between hopping frequency and performance overhead should be considered in the actual situations, which remains to be further discussed in our future work.

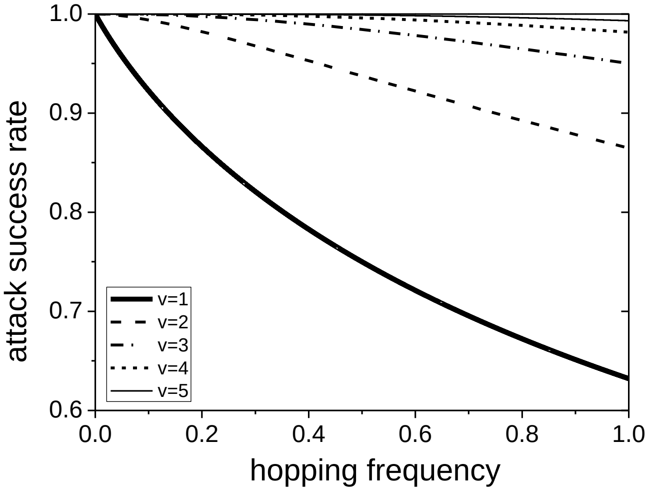

Figure 4. The ASR vs the Normalized Hopping Frequency

5. Conclusions

Our work has focused on modeling the effectiveness of port hopping against the reconnaissance attack, which is the first attack phase of various network attacks. We have introduced probabilistic models based on urn models to quantify the security performance of port hopping in terms of the port pool size, number of probes, number of vulnerable services, and the hopping frequency. The results of theoretical analysis shows that port hopping improves the effectiveness in thwarting the reconnaissance attacks, and the perfect port hopping reduces the likelihood of attack success by 37% as compared to using static addresses. Moreover, the results also shows that port hopping is an acceptable defense if there is a small number of vulnerable services and a large port pool size.

In the actual network, a successful cyber attack contains a series of attack phases other than reconnaissance, such as gain access, exploit vulnerabilities and escalate privileges *et al.*, port hopping as a proactive cyber defense mechanism will also be useful to mitigate these attack phases. Consider an integrated attack that is comprised of *i* attack phases, and the attacker needs to connect the target *j* times in each phase, if each of the attack phases has an identical success rate *r*, then the likelihood of the integrated attack success is r^{i*j} under perfect port hopping. If $r = 0.99$ ($v = 5$), $i = 5$, and $j = 8$, then $r^{i*j} = 0.99^{40} \approx 0.35$, which is too small to succeed. Therefore, port hopping is an effective and promising proactive defense technology in thwarting cyber attacks.

Acknowledgements

This work is funded by Research Fund for the Doctoral Program of Higher Education of China (20114307110006) “Network QoS Guarantee Technology based on Virtualization”; National Key Basic Research Program of China (973 Program) “Reconfigurable Network Emulation Testbed for Basic Network Communication” (2012CB315906); Research on Trustworthy Cross-Domain Information Exchanging Technology Based M-TCM(KJ-12-07).

References

[1] J. Mirkovic and P. Reiher, “A Taxonomy of DDoS Attack and DDoS Defense Mechanisms”, ACM SIGCOMM Computer Communication Review, vol. 34, no. 2, **(2004)**, pp. 39-53.

[2] N. Baik and N. Kang, “Experimental Study of DDoS Defense System for Web Service”, International Journal of Security and Its Applications, vol. 7, no. 5, **(2013)**.

[3] J. Wang, Z. Wang, and D. Kui, “A Network Intrusion Detection System based on Artificial Neural Networks”, Proceedings of the 3rd ACM International Conference on Information Security, **(2004)**, pp. 166-170.

[4] L. Tan and T. Sherwood, “A High Throughput String Matching Architecture for Intrusion Detection and Prevention”, Proceedings of the 32nd International Symposium on Computer Architecture, **(2005)**, pp. 112–122.

[5] T. Yan and Y. F. Zhang, “For Deformation Web Attacks based on Feature Recognition IPS Intrusion Prevention Technology Research”, International Journal of Future Generation Communication and Networking, vol. 7, no. 2, **(2014)**.

[6] Y. Zhang and Z. Li, “Overview on Moving Target Defense Technology, Communications Technology”, vol. 6, **(2013)**, pp. 111-113.

[7] M. Carvalho and R. Ford, “Moving-target defenses for computer networks”, IEEE Security & Privacy, vol. 12, no. 2, **(2014),** pp. 73–76.

[8] National Cyber Leap Year Summit 2009 Co-Chairs Report. Tech. Rep. Available: http://www.ncbi.nlm.nih.gov/docs/, **(2009)**.

[9] H. Wang, Q. Jia, D. Fleck, W. Powell and F. Li, “A Moving Target DDoS Defense Mechanism”, Computer Communications, vol. 46, **(2014)**, pp. 10–21.

[10] E. Al-Shaer, “Toward Network Configuration Randomization for Moving Target Defense [M]”, Moving Target Defense, Springer New York, vol. 54, **(2011)**, pp. 153-159.

[11] M. Atighetchi, P. Pal, F. Webber, and C. Jones, “Adaptive Use of Network-Centric Mechanisms in Cyber-Defense”, Proceedings of 6th IEEE Int”l Symp. Object-Oriented Real-Time Distributed Computing, **(2003)**, pp. 183–192.

[12] H. C. J. Lee and V. L. L. Thing, “Port Hopping for Resilient Networks”, Proceedings of IEEE 60th Vehicular Technology Conference, vol. 5, **(2004)**, pp. 3291–3295.

[13] G. Badishi, A. Herzberg, and I. Keidar, “Keeping Denial of Service Attackers in the Dark”, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3724, **(2005)**, pp. 18–32.

[14] G. Badishi, A. Herzberg and I. Keidar, “Keeping Denial of Service Attackers in the Dark”, IEEE Transactions on Dependable and Secure Computing, vol. 4, no. 3, **(2007)**, pp. 191–204.

[15] L.Y. Shi, C. F. Jia, S.W L$\ddot{u}$, and Z. H. Liu, “Port and Address Hopping for Active Cyber-Defense”, Lecture Notes in Computer Science, vol. 4430, **(2007)**, pp. 295–300.

[16] Y. B. Luo, B. S. Wang and G. L. Cai, “Effectiveness of Port Hopping as a Moving Target Defense”, Proceedings of the 7th International Conference on Security Technology (SecTech-2014), **(2014)**.

[17] N. Hoque, M. H. Bhuyan, R. Baishya, D. Bhattacharyya and J. Kalita, "Network Attacks: Taxonomy, Tools and Systems", Journal of Network and Computer Applications, vol. 40, **(2014)**, pp. 307–324.

[18] Z. Fu, M. Papatriantafilou, and P. Tsigas, "Mitigating Distributed Denial of Service Attacks in Multiparty Applications in the Presence of Clock Drifts", Proceedings of 27th IEEE International Symposium on Reliable Distributed Systems (SRDS-2008), **(2008)**, pp. 63–72.

[19] K. Hari and T. Dohi, "Sensitivity Analysis of Random Port Hopping", Proceedings of the 2nd International Symposium on Multidisciplinary Emerging Networks & Systems (MENS-2010), **(2010)**, pp. 316–321.

[20] K. Hari and T. Dohi, "Dependability Modeling and Analysis of Random Port Hopping", Proceedings of the 9th International Conference on Ubiquitous Intelligence and Computing and 9th International Conference on Autonomic and Trusted Computing, **(2012)**.

Authors

Yue-Bin Luo, he received his B.E. degree from Tianjin University (TJU), China, in 2010, and received his M.S. degree from National University of Defense Technology (NUDT), China, in 2012. Now, he is a Ph.D. candidate in NUDT. His research interests include intrusion detection, computer networks and information security, Moving Target Defense (MTD).

Bao-Sheng Wang, he is now professor, doctoral supervisor in computer networks and communication of National University of Defense Technology (NUDT). His research interests include network security, computer network architecture and network protocol technology.

Gui-Lin Cai, she received her B.E. and M.S. degree from National University of Defense Technology (NUDT), China, in 2005 and 2010. Now, she is a Ph.D. candidate in NUDT. Her research interests include computer networks and information security.