Cyber Security Enhancement via Software-Defined Pseudo-Random Private IP Address Hopping

Andre Slonopas, Warren Thompson, Zona Kostic

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf

***Abstract*—**Obfuscation is one of the most useful tools to prevent network compromise. Previous research focusedon the obfuscation of the network communications between external-facing edge devices. This work proposes the use of two edge devices, external and internal facing, which communicates via private IPv4addresses in a software-defined pseudo-random IP hopping. This methodology does not require additional IP addresses and/or resources to implement. Statistical analyses demonstrate that the hopping surface must be at least 1e3 IP addresses in size with a broad standard deviation to minimize the possibility of coincidence of monitored and communication IPs. The probability of breaking the hopping algorithm requires a collection of at least 1e6 samples, which for large hopping surfaces will take years to collect. The probability of dropped packets is controlled via memory buffers and the frequency of hops and can be reduced to levels acceptable for video streaming. This methodology provides an impenetrable layer of security ideal for information and supervisory control and data acquisition systems.

***Keywords*—**Moving Target Defense, cybersecurity, network security, hopping randomization, softwaredefined network, network security theory.

I. Introduction

RECENT years introduced an exponential spur in the use and applications of computer networks [1], [2]. Information traffic forecasts predict even a more dramatic increase in the near future [1], [3]. The extensive interconnectivity simplified the information management for users, inevitably decentralizationof information into the cyber domain introduced a plethora of vulnerabilities. Interconnected devices, by their core design, are conceived to collect, analyze, monitor and present information to the user.

Confidentiality, integrity, security and availability of the data, however, is often treated as anafterthought [4], [5].

IP hopping has been introduced as a potential additional layer of security to the information systems [6], [7]. More specifically, the IP hopping has enabled the moving target defense (MTD) concepts to be leveraged as a protection mechanism. The ever-changing IP addresses prevent the intruder from successfully finding the address of a device in interest [8], [9]. This methodology, however, requires numerous public IPs thus adding to the cost and coordination complexity among the key holders [7]-[9]. Furthermore, in order for the hopping mechanism to become an effective countermeasure a fairly large IP spectrum needs to be acquired [10], [11]. This IP space needs to be properly randomized with a unidirectional algorithm [12], [13]. As an additional layer of protection, the IP randomization must be dynamic [14], [15]. Effective techniques for dynamic randomization have been demonstrated by leveraging Software Defined Networks (SDN) [16], [17]. If, however, the IP space is not adequately sized even most dynamic randomization algorithms can be elucidated via conventional heuristic techniques. The IP size requirement makes the MTD economically unfeasible for majority of the users. Considering the above discussion, IP hopping is a powerful protection mechanism if the associated costs and need for specialized equipment can be reduced or eliminated.

This paper presents a concept of an interconnected external and internal facing end nodes that communicate using private IPv4 addresses in a pseudo-random IP hopping algorithm. Conventional obfuscation methods, such as The Onion Routers (ToR), create external traffic concealment; whereas the methodology proposed in this paper provides an internal convolution. Randomization of hops is implemented via software on the end nodes. The approach described in this paper does not require specialized hardware and can be implemented on any size system. Furthermore, use of private IPv4 space allows for a very large hopping surface at essentially no additional cost and would be outside of a monitoring capability of nearly all malicious actors. Single Class A (i.e. 10.0.0.0/24) network IP space is leveraged to give a larger surface for the statistical analysis; the concept, however, can easily be applied to any IPv4 private class domain. Statistical analysis on malicious data and fuzzing injection based on hopping surface and bot-net size is analyzed. Bot-net requirements to survey 500,000 is unreasonable for majority of malicious users, yet the 500,000 private IP addresses make up slightly less than 0.02% of Class A private network. IP hopping could also span other private IP Classes, which would make networkscanning and monitoring beyond reach of even nation states. Packet loss based on frequency of hops are found to be negligible and reduced to video streaming levels if the hopping frequency is reduced to above 0.2 Hz and a memory buffer is introduced. Probability to break the hopping algorithm is also evaluated. Hopping algorithm can remain unbroken with regular updates.

II. Theoretical Concept and Computational Details

Conceptual high-level layout of the analyzed network is shown in Fig. 1. The operability of the network is contingent on the accurate data acquisition from the Remote Terminal Units

Andre Slonopas* and Zona Kostic are with School of Science, Technology, Engineering, and Math, American Public University System, 111 W.Congress St., Charles Town, WV, 25414 (*e-mail: aslonopas@apus.edu).

Warren Thompson is with Cyber Operations, Dakota State University, 820 N Washington Ave. Madison, SD 57042, USA.

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf

(RTUs) and analysis by the control unit. The control unit (CU) is a simplified version of a Supervisory Control and Data Acquisition (SCADA) system. The methodology of this paper seeks to provide a general concept fora MTD via IP hopping in private space methodology with a broad range of applications. The key elementof this novel approach are the two edge routers which communicate via pseudo-random software defined IP hopping in the IPv4 private space. In the schematic of Fig. 1, the authorized user is on the inside of the Local Area Network (LAN) and within the LAN of the CU. In reality, however, authorized user may be connecting to Router 2 via a Wide Area Network (WAN). In the latter case the authorized user will need to know the IP hopping algorithm between the two routers. Since the IP hopping mechanism is software defined, it can easily be provided to the user. It is assumed that the malicious actor does not have physical access to the hardware and is conducting reconnaissance and data injection via WAN.

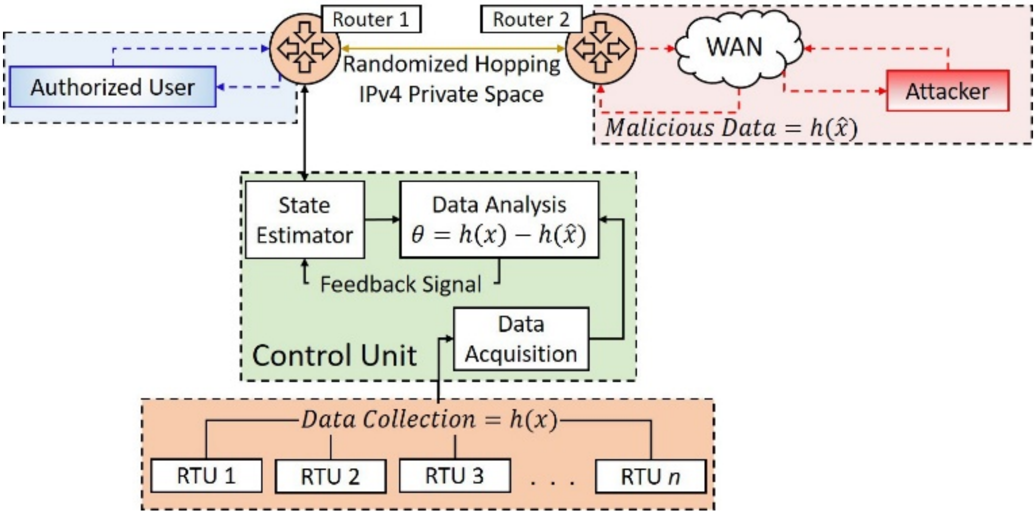

Fig. 1 Schematic of the cyber security enhancement via private IP address hopping between twoedge routers

The statistical model is based on three assumptions and analogous to the Bayes cross-validation leveraged in machine learning [18], [19]. First assumption is that the a priori probabilities, denoted P_1 and P_2, are N dimensional and found on real Euclidean space. These probabilities encompass all of the possible external influences that can affect the outcome before any action is taken. Second assumption is a simple binary hypothesis, H_0 and H_1. The two hypotheses correspond to malicious actor's IP addressbeing different and identical respectively to the IP address during a hop. Hypotheses correspond to the N observations, such that they can be represented by $\theta\theta$ vector in an N-dimensional space, i.e. $\theta\theta \triangleq [\theta\theta_1, \theta\theta_2 \cdots \theta\theta_N]$. Third assumption presumes four possible courses of action: C_{00}, C_{01}, C_{10}, C_{11}. The subscripts denote chosen and true hypotheses respectively. Given the above discussion, the greatest interest is in the course of action where the chosen and true hypotheses correspond to identical IP addresses of the malicious actor and the router, i.e. C_{11}. The expectation equation can thus be writtenby averaging over the a priori probabilities and the probability that a certain course of action will be taken [20]:

$$\mathcal{R} = E\{C\} = \sum_{i=0}^{1}\sum_{j=0}^{1} C_{ij}\Pr\left(I_i H_j\right) \tag{1}$$

where I_{ii} and H_{ii} represent successful injection and assumed hypothesis respectively. Given the two possible hypotheses, H_0 and H_1, the observation space, i.e. private IP space, can be broken into two parts, Z_0 and Z_1 respectively. The total observation space is $Z = Z_1 + Z_0$. The associated risk can thus be analyzed in the decision regions rewritten in terms of transition probabilities:

$$\begin{aligned}\mathcal{R} = {} & C_{00}P_0\int_{Z_0} p_r|H_0(\mathcal{R}|H_0)d\mathcal{R} + C_{01}P_1\int_{Z_0} p_r|H_1(\mathcal{R}|H_1)d\mathcal{R} \\ & + C_{10}P_0\int_{Z-Z_0} p_r|H_0(\mathcal{R}|H_0)d\mathcal{R} \\ & + C_{11}P_1\int_{Z-Z_0} p_r|H_1(\mathcal{R}|H_1)d\mathcal{R}\end{aligned} \tag{2}$$

Observing that the probability of hypothesis H_0 and H_1 on an entire observation space is equal to 1,(2) can thus simplify to:

$$\begin{aligned}\mathcal{R} = {} & \int_{Z_0}\Bigg[\underbrace{\left(P_1(C_{01} - C_{11})p_r|H_1(\mathcal{R}|H_1)\right)}_{Term\ 1} \\ & - \underbrace{\left(P_0(C_{10} - C_{00})p_r|H_0(\mathcal{R}|H_0)\right)}_{Term\ 2}\Bigg]dR \\ & + C_{10}P_0 + C_{11}P_1\end{aligned} \tag{3}$$

It is assumed that negative probabilities and course of actions are not realistic, thus only positive valuesinside of the integral are analyzed. Values where second term is larger than the first are included in Z_0, and values where the second term is smaller than the first contribute to Z_1. Thus, successful injection of malicious data packets occurs in the case where:

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf

$$R = P_1(C_{01} - C_{11}) p_r|H_1(R|H_1) \geq P_0(C_{10} - C_{00}) p_r|H_0(R|H_0) \tag{4}$$

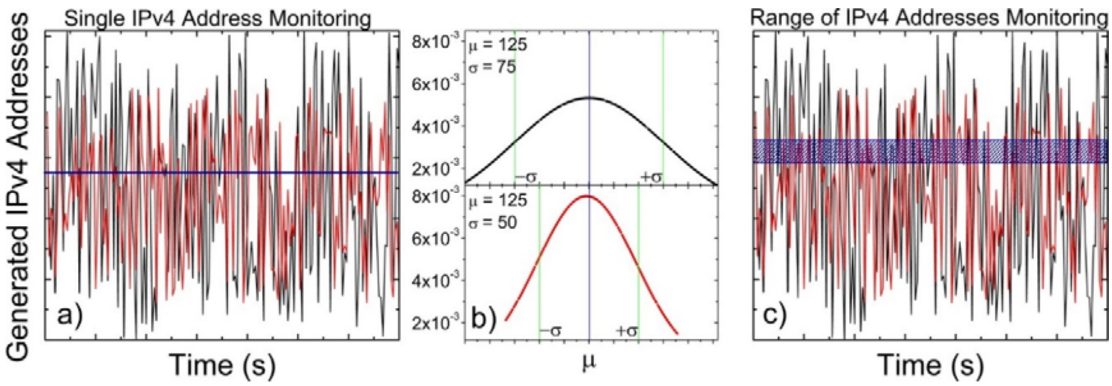

Fig. 2 (a) and (c) randomly generated IPv4 addresses with the standard deviations shown in (b). Single and range of IPv4 address monitoring is denoted by the blue line and box in (a) and (c) respectively

Rearranging (4) the likelihood of a malicious data injection is denoted as:

$$\Lambda(\mathcal{R}) \triangleq \frac{p_r|H_1(\mathcal{R}|H_1)}{p_r|H_1(\mathcal{R}|H_1)} \tag{5}$$

It is assumed that the routers communicate only via a single channel on a single IP address. Malicious actor, however, can monitor a single or a spectrum of IP addresses, Figs. 2 (a) and (b), respectively. IP addresses were randomly generated giving N samples. IP samples are Gaussian with an independent random data transfer of $h(x)$ and variance of σ^2. For simplicity, mean, μ, for the data was chosen to be 125, which is very close to the expected mean of 255 bits of 122.5. Standard deviation of the data was varied for the analysis. Randomly generated IPv4 addresses based on differing standard deviations are highlighted in Fig. 2.

Under H_0 no malicious data injection is possible. Under H_1 data transfer is equal to the summation of legitimate and malicious data, i.e. $h(x) + h(\hat{x})$. For Gaussian sampling, the probability density of $\theta\theta_i$ for H_0 and H_1 becomes:

$$p_{\theta_i|H_0}(\mathcal{R}_i|H_0) = p_{h(x)_i}(\mathcal{R}_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left[-\frac{\mathcal{R}_i^2}{2\sigma^2}\right]}$$
and
$$p_{\theta_i|H_1}(\mathcal{R}_i|H_1) = p_{h(x)_i + h(\hat{x})_i}(\mathcal{R}_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left[-\frac{(\mathcal{R}_i - h(\hat{x}))^2}{2\sigma^2}\right]} \tag{6}$$

respectively. In the case where a malicious actor monitors a range of IPs, due to the statistical independence of $h(x)$, the joint probability density is simply the multiplication of individual probability densities. Substituting probability densities into the likelihood of data injection equation and simplifyinggives the basis for the statistical analysis as:

$$\ln \Lambda(\mathcal{R}) = \frac{h(\hat{x})}{\sigma^2} \sum_{i=1}^{N} \mathcal{R}_i - \frac{N h(\hat{x})^2}{2\sigma^2} \tag{7}$$

Packet loss for a hopping system was calculated assuming an uncorrelated system, i.e. Hurst exponentof 0.5, and a constant effective bandwidth, i.e. lim $B_{eeee}(\theta\theta, t) = Constant$. Thus, the probability of $t \to \infty$ packet loss can be estimated by [21], [22]:

$$log(P_{loss}) = -\frac{2\big(c - \mu(k)\big)}{\sigma^2(k)} \log(e) - log\left[\frac{2\mu(k)(c - \mu(k))}{\sigma^2(k)}\right] \tag{8}$$

where $\mu(k)$ and $\sigma^2(k)$ are the traffic mean and variance computed from the packet input rate $\alpha(k)$:

$$\mu(k) = \frac{1}{N(t)} \sum_{i=0}^{N(t)-1} \alpha(k-i)$$
and
$$\mu(k) = \frac{1}{N(t)-1} \sum_{i=0}^{N(t)-1} [\alpha(k-i) - \mu(k)]^2 \tag{9}$$

k denotes the time interval between hops, and (t) is the number of time intervals used for thecalculation of the mean and the variance.

For the analysis, 10,000 random IP addresses were generated. Data transfer speeds were assumed to be the typical 100 Mbps, or 12.5 MBps. Malware size was assumed to be the average reported from 2008 of 338 kB, which equates to ~3% of the bandwidth [23]. Thus, it is assumed that a single concurrence of IP addresses will result in a successful injection of the malicious code. Functionality of such malware, however, will be limited. Furthermore, communication and data exfiltration will not be possible until the next IP address concurrence between malicious actor and the internal router.

III. Results and Discussion

Probability of IP matching versus the number of possible IPs and the standard deviation of the IP hopping surface is shown in Fig. 3 (a). Standard deviation was computed from the possible values in the 4th quartet of an IP address and manually manipulated for the purpose of the analysis. Probability of matching IPs is significant, i.e., ~10%, for hopping surfaces below 1,000 IPs. Furthermore, in these cases, the standard deviation does not play a significant role in reducing the probability of H_1. Standard deviation, however, plays a significant role when the IP hopping surface increases beyond 1,000 IPs which decreases H_1 by orders of magnitude. Thus, the pseudo-random IP generator should be designed with large σ at its core. Cases where a range of IPs is monitored by the malicious user, the probability of H_1 is simply scaled to the probabilities of a single channel monitoring, Fig. 3 (b). As an example, for a case of 1,000 IPs with a large σ, malicious user would have to monitor 10 IP addresses to achieve ~2% probability of H_1. In this case approximately every 50th hop will result in a matching IP set. Increasing hopping surface to 2,500 IPs, decreases the possibility of $H1$ to ~8.5*10^{-4}, where every 117,000th hop willproduce a match. Assuming that the hops occur every 5 seconds a matching IP set will occur approximately every 7 days. A malicious user is required to maintain a constant communication with the device for an effective attack. In the case of 2,500 IP hopping surface, malicious user would be forced to monitor over 2,000 IP addresses. The effort required to maintain such an extensive monitoring capability would be beyond the reach of majority of malicious users. Furthermore, hopping surface of 2,500 IPs makes up less than 0.02% of the Class A private IPv4 space and can easily be scaled by orders of magnitude.

Probability of packet loss versus hopping times is shown in Fig. 4 (a). Probability of a packet loss in the initial 25 seconds is commensurate and independent of the hopping frequency. Furthermore, the statistical probability of packet loss in the initial 20-25 seconds is excessively high and is attributed to insufficient data for an accurate analysis. Thus, only the data after the initial 30 seconds is considered forthe discussion. Hopping frequency plays a significant role in the probability of dropped packets. The more frequent hops result in a greater probability of the dropped packets. This behavior would be expected in the cases where no memory buffer is present, since the packets that were not sent out priorto the hop will be dropped. The mitigation strategy to minimize the probability of a drop would be to implement a memory buffer for the unsent packets. Statistical analysis of the probability of a drop for a 3 second hopping interval with various buffer sizes is shown in Fig. 4 (b). Significant reduction in the probability of drops is observed even with small memory buffers of 1,500 bytes, which would equate to approximately one packet. The probability of drops decreases even further with larger memory buffers, however, with diminishing improvements. Memory buffers of 1,500 bytes are sufficient to reduce probability of drops to acceptable levels for most applications. Packet loss drops of 1% are acceptable for video streaming and are achievable in the hopping scheme with a 3,000 byte buffers. The oscillatory behavior in the probability of packet drops at 30 and 45 seconds is attributed to the $\mu(k)$ and $\sigma^2(k)$ from the random number generator.

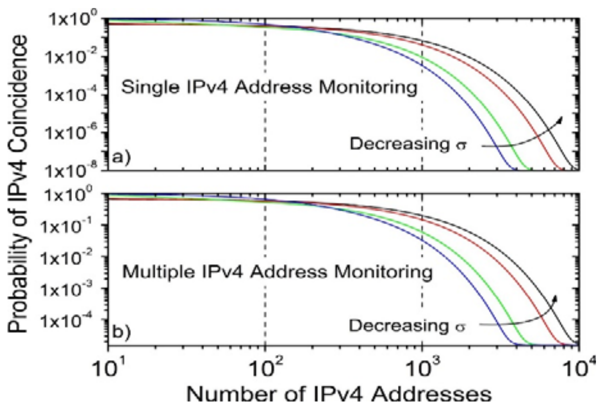

Fig. 3 (a) Probability of IPv4 coincidence versus the size of the hopping surface for a single monitored IP address, (b) Probability of IPv4 coincidence versus the size of the hopping surface for 10 monitored IPaddresses

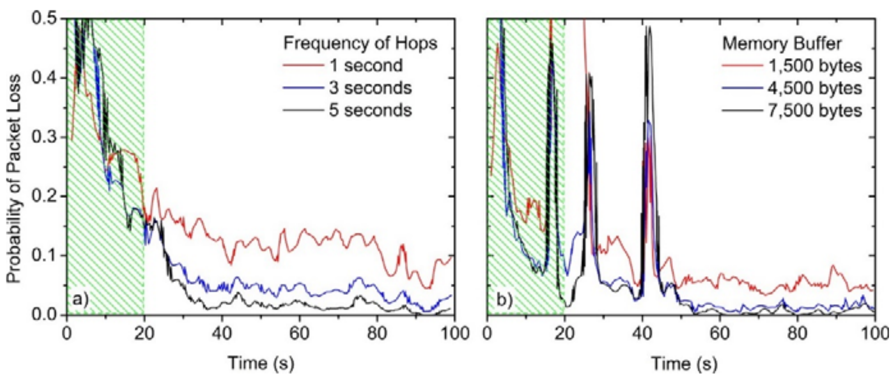

Fig. 4 (a) Probability of packet loss as a function of hopping frequency with no memory buffer in place, (b) Probability of packet loss for a 3 second hopping frequency for various size memory buffers.Areas highlighted in green are based on insufficient amount of data generated from random numbergenerator and are thus omitted from the discussion

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf

Probability of breaking the hopping algorithm was analyzed statistically using the Sigmoid Analysis. Analysis was conducted as a function of (k) and $\sigma^2(k)$ versus the number of samples, Figs. 5 (a) and (b). respectively. Heat maps demonstrate a nonlinear relationship between the mean, standard deviation and the number of samples required to elucidate the hopping algorithm. Nearly 1e6 samples would have to be collected for a reasonable chance of elucidating the hopping scheme. Furthermore, malicious actor would need to monitor more than one IPv4 address, be aware of the embedded hopping scheme, and monitored IPv4 addresses would need to coincide with the addresses used during the hop. Using large hopping surface and $\sigma^2(k)$ in a private IP software defined hopping scheme which can be altered regularly makes the compromise of such system nearly impossible.

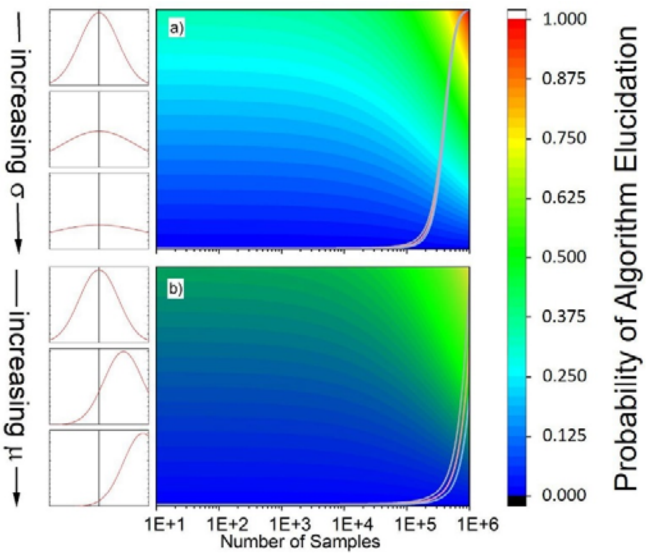

Fig. 5 (a) Probability of hopping algorithm elucidation versus the number of samples, static $\mu(k)$ and dynamic $\sigma^2(k)$, (b) Probability of algorithm elucidation as a function of $\mu(k)$. Black line in the subgraphsdenotes monitored IPv4 address. $\mu(k)$ and $\sigma^2(k)$ are found to play a cardinal role in the probability of breaking the hopping algorithm and need to be considered in algorithm development

IV. Conclusion

This paper presented a software based cyber security concept by leveraging two edge routers for software defined private IP address hopping mechanism. Conceptual diagram and the hopping algorithm are discussed briefly, however, left to be defined for the developers. Probability of IP coincidence is analyzed for a single and multiple monitored IP addresses. It is found that the coincidenceof IP addresses is less likely in a broader hopping surface. It is also found that the hopping surfaces should be at least 1e3 IPs in size. This number, however, constitutes only a fraction of a percent of the Class A private IP space. Probability of packet loss versus frequency of hops and memory buffer demonstrates that hopping frequencies should be below 0.2 Hz. Memory buffers of at least 1 packet sizewill reduce packet drops to thresholds acceptable for video streaming. Probability of breaking the hopping algorithm is also analyzed. Probability of breaking the hopping algorithm is virtually zero below 1e6 samples. Methodology discussed in this paper provides a moving target defense that is virtually impenetrable yet operates in a private IP space requiring no additional resources. Furthermore, this methodology is a proactive approach against data injection, protection from DOS attacks, and provides an extra layer of security ideal for information control systems.

References

[1] F.-J. Muro, N. Skorin-Kapov and P. Pavon-Marino, "Revisiting core traffic growth in the presence of expanding CDNs," Computer Networks, vol. 154, pp. 1-11, 2019.

[2] R. Malik, "Spread spectrum-secret military technology to 3G," IEEE History of Telecommunications Contest, 2001.

[3] J. Qingmin, X. Renchao, H. Tao, L. Jiang and L. Yunjie, "The Collaboration for Content Delivery and Network Infrastructures: A Survey," IEEE Access, vol. 5, pp. 18088 - 18106, 2017.

[4] G. Gan, Z. Lu and J. Jiang, "Internet of Things Security Analysis," International Conference on Internet Technology and Applications, pp. 1 - 4, 2011.

[5] L. Shi, C. Jia, S. Lu and Z. Liu, "Port and address hopping for active cyber-defense," Intelligence and Security Informatics, PAISI, vol. 4430, pp. 295-300, 2007.

[6] S.-Y. Chang, Y. Park and B. B. Ashok Babu, "Fast IP hopping randomization to secure hop-by-hop access in SDN," IEEE Transactions on Network and Service Management, vol. 16, no. 1, pp. 308 - 320, 2019.

[7] A. Teixeira, G. Dan, H. Sandberg and K. H. Johansson, "A cyber security study of a SCADA energy management system: stealthy deception attacks on the state estimator," IFAC Proceedings Volumes, vol. 44, no. 1, pp. 11271 - 11277, 2011.

[8] K. Zheng, X. Zhao, X. Li and Y. Zhou, "A SDN-based IP address hopping method design," Proceedings of the 2016 5th International Conference on Measurement, Instrumentation and Automation (ICMIA 2016), 2016.

[9] M. Marx, M. Schwarz, M. Blochberger, F. Wille and H. Federrath, "Context-Aware IPv6 Address Hopping," Information and Communications Security. ICICS, vol. 11999, pp. 539 - 554, 2019.

[10] S.-Y. Chang, Y. Park and A. Muralidharan, "Fast address hopping at the switches: Securing access for packet forwarding in SDN," NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pp. 454 - 460, 2016.

[11] C. Zhao, C. Jia and K. Lin, "Technique and application of end-hopping in network defense," First ACIS International Symposium on Cryptography, and Network Security, Data Mining and Knowledge Discovery, E-Commerce and Its Applications, and Embedded Systems, pp. 266-270, 2010.

[12] M. Atighetchi, P. Pal, F. Webber and C. Jones, "Adaptive use of network-centric mechanisms in cyber-defense," Sixth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, pp. 183 - 192, 2003.

[13] M. Dunlop, S. Groat, W. Urbanski, R. Marchany and J. Tront, "MT6D: A Moving Target IPv6 Defense," MILCOM 2011 Military Communications Conference, pp. 1321 - 1326, 2011.

[14] D. L. Kewley, R. A. Fink, J. Lowry and M. Dean, "Dynamic approaches to thwart adversary intelligence gathering," Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01, pp. 176 - 185, 2001.

[15] D. E. Broth and R. E. Ziemer, Introduction to Spread-spectrum Communications, Englewood Cliffs, NJ: Prentice Hall, 1995.

[16] J. Haadi Jafarian, E. Al-Shaer and Q. Duan, "Random host mutation for Moving Target Defense," International Conference on Security and Privacy in Communication Systems, vol. 106, pp. 310 - 327, 2012.

[17] P. Kampanakis, H. Perros and T. Beyene, "SDN-based solutions for Moving Target Defense network protection," Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, pp. 1 - 6, 2014.

[18] S. Watanabe and M. Opper, "Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory," Journal of Machine Learning Research, vol. 11, pp. 3571-3594, 2010.

[19] A. Vehtari, A. Gelman and J. Gabry, "Practical Bayesian model

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf

evaluation using leave-one-out cross-validation and WAIC," Statistics and Computing, vol. 27, pp. 1413-1432, 2017.

[20] R. Harman and M. Prus, "Computing optimal experimental designs with respect to a compound Bayes risk criterion," Statistics & Probability Letters, vol. 137, pp. 135-141, 2018.

[21] D. Zhang and D. Ionescu, "Reactive estimation of packet loss probability for IP-based video services," IEEE Transactions on Broadcasting, vol. 55, no. 2, pp. 375-385, 2009.

[22] D. Zhang and D. Ionescu, "A new method for measuring packet loss probability using a Kalman filter," IEEE Transactions on Instrumentation and Measurement, vol. 58, no. 2, pp. 488-499, 2009.

[23] Z. M. Shafiq, S. A. Khayam and M. Farooq, "Embedded malware detection using Markov n-Grams," Detection of Intrusions and Malware, and Vulnerability Assessment. DIMVA 2008. Lecture Notes in Computer Science., vol. 5137, 2008.

Open Science Index, Computer and Information Engineering Vol:16, No:5, 2022 publications.waset.org/10012534.pdf