# On Evolutionary Computation for Moving Target Defense in Software Defined Networks

Adetokunbo Makanju
KDDI Research
2-1-15 Ohara
Fujimino-shi, Saitama, Japan 356-8502
to-makanju@kddi-research.jp

A. Nur Zincir-Heywood
Dalhousie University
6050 University Avenue
Halifax, NS, Canada B3H 4R2
zincir@cs.dal.ca

Shinsaku Kiyomoto
KDDI Research
2-1-15 Ohara
Fujimino-shi, Saitama, Japan 356-8502
kiyomoto@kddi-research.jp

## ABSTRACT

Moving Target Defense (MTD) is a paradigm in cyber-security that proposes cyber-systems that continually change their configurations to avoid easy surveillance from adversaries. Despite its promise, MTDs have yet to be fully adopted in real world systems. Using the concepts from the general theory of MTD, this paper proposes that the goal of MTDs i.e continually adapting to their environment to evade attack is similar to the concept of an organism evolving to survive in its habitat. This implies that Evolutionary Computation (EC) techniques could be used to fulfill the goal of MTD, specifically in relation to intrusion detection in the emerging field of Software Defined Networks (SDNs). The programmable nature of SDNs provide a degree of flexibility not possible in conventional networks. Thus, making it possible to completely automate the configuration of such networks.

## CCS CONCEPTS

**•Security and privacy → Network security; •Computing methodologies → Genetic algorithms;**

## KEYWORDS

Genetic Algorithms, Software Defined Networks, Moving Target Defense

**ACM Reference format:**
Adetokunbo Makanju, A. Nur Zincir-Heywood, and Shinsaku Kiyomoto. 2017. On Evolutionary Computation for Moving Target Defense in Software Defined Networks. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 2 pages.
DOI: http://dx.doi.org/10.1145/3067695.3075604

## 1 INTRODUCTION

Modern cyber systems are very static, this property makes them easy to manage. However, this also makes them very vulnerable to attacks. Attackers spend a lot of time gathering information about their targets. The information gathered remains valid until the network configuration is changed. Moving Target Defense (MTD) mitigates this problem by proposing a scenario where the configuration of cyber systems constantly change [6]. This means that an the attacker must gather information and execute the attack before the configuration changes.

The MTD approach has yet to be fully adopted at the network layer in real world systems due to the significant overheads that may be required, especially if it has to be performed frequently. This problem can be exacerbated if the search space of possible configurations is small. When the search space is small, an attacker can easily guess the new configuration. The only way to mitigate this is to move frequently, which can be disruptive.

Software Defined Networks (SDN) are networks that can be designed , built and operated in a highly scalable and adaptable manner [3]. The simplest analogy is to view SDNs as an extension of the virtualization of server infrastructure to the network infrastructure. Due to their programmability, SDNs can more readily be changed as the needs of users of the network changes at minimal cost.

The flexibility and programmability of the control plane that SDNs bring to the table also has an impact on the kind of security solutions that can be deployed on them. SDNs allow many aspects of the network infrastructure to be changed programmatically. This means that MTD can be more easily implemented and the search space of possible configurations is now much larger.

Moving Target Defense (MTD) in SDNs is not new, there is prior work in this area [2, 5]. While the large space afforded by SDNs has its advantages, it can also be a disadvantage. It can lead to issues of scalability in finding a workable configuration in cases where a random configuration may not work. Evolutionary Computation (EC) algorithms are designed to efficiently search large spaces for optimal solutions are therefore good candidates for use in MTD in SDN. While EC has been applied to the problem of MTD [4] , to the best of our knowledge, there is no significant work that applies it in SDN systems.

## 2 METHODOLOGY

The framework for the general MTD process as described by Zhuang et. al. in [6] is shown in Fig. 1. The process starts with the deployment of the system. From time to time the MTD system will choose a new configuration for the system taking into consideration its environmental information, which may include execution status and intrusion alerts. Each new configuration needs to be tested against the policies of the system e.g. Service Level Agreements (SLAs), enterprise network policies etc., to ensure that none of the policies are violated. If a policy is violated a new configuration must be chosen. This process repeats until an appropriate configuration is found.


*GECCO '17 Companion, Berlin, Germany*

DOI: http://dx.doi.org/10.1145/3067695.3075604

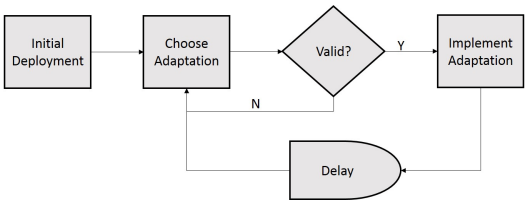


**Figure 1: The General MTD Process [6]**

This basic configuration raises a number of issues that Zhuang et. al. [6] describe as the three essential problems of MTD systems. These are how to choose another configuration (MTD problem), how to plan the configuration change i.e. what step-by-step adaptations lead to next configuration (Adaptation Selection Problem) and when to carry out the adaption (Timing problem).

We also note that the timing problem goes beyond the definition in [6], we believe that it is a mix between the scalability of the configuration selection (size of the search space), frequency of adaptation (as mentioned by Zhuang et. al.) and the time required for an attacker to gather enough information for an attack.

For an MTD system to be of practical use the search space of possible configurations needs to be large enough for an an attacker not to easily guess the next configuration i.e the MTD Entropy hypothesis [6]. Unfortunately this is also detrimental to the defender as it means that searching for a valid configuration will be time consuming. Then there could be scenarios where the validity of configuration is not a binary choice $[Yes, No]$ but lies in a continuous range $[0 - 1]$ that defines the degree to which it meets the requirements and we can only select configurations that meet requirements beyond a certain threshold. All of these scenarios add to complexity of the formulation and has made MTD impractical for certain applications.

We however argue that the three MTD problems cited by Zhuang et. al. [6] are typical of most EC problems, therefore EC is an appropriate approach for handling MTD in general.

Our proposed EC approach to MTD starts with an initial deployment as shown in Fig. 2. The configuration of the system can be coded to represent an individual's genotype as a whole or broken down into its constituent parts, where each part represents a different kind of individual (species) in the population. The latter model is preferred as it reduces the complexity of the solution that has to be evolved. The final solution in such a scenario will be a combination of the different individuals of each specie which have the best fitness.

The deployed system can change its configuration both proactively i.e. after a fixed or random length of time or re-actively i.e. in the case of extraordinary circumstances e.g. an attack, system fault, policy change e.t.c.. The system would also have a representation of its environment information i.e. system policies, systems goals (SLAs) e.t.c. Together the environment information and any existential trigger events form the landscape to which individuals in the population must adapt. So the fitness function would be designed around these parameters.

## 3 CONCLUSIONS

Due to the large configurable search space they make available, SDNs are better candidates for MTD-based security solutions than

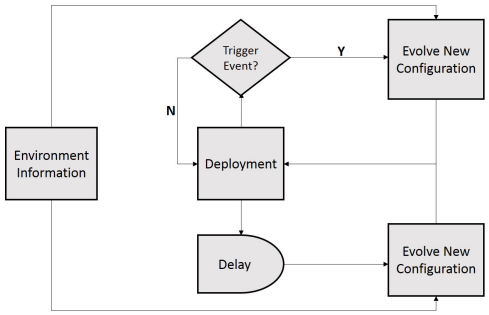


**Figure 2: An EC approach to MTD**

conventional networks. This large search space also has a hidden advantage in that the network configuration does not have to change frequently. The amount of time required by the attacker to gather information about the new configuration or guess the new configuration is sufficiently large enough to achieve the aim of preventing attacks, while not requiring frequent configuration changes that can be very disruptive. Despite the good fit, EC approaches have not received adequate attention in the implementation of MTD solutions for SDNs. The initial problem to be overcome in the use of EC for MTD solutions is the representation of the individuals in the population. The work carried out by Zhuang et. al. in [6], goes a long way in solving this problem. It provides an excellent starting point for any work in this area. In these works the authors provide a complete theoretical framework for MTD and cyber attacks, with formal definitions for all of the actors in the system, which can be used to come up with appropriate representations for individuals in the population. As the optimization problem to be solved is likely to involve multiple objectives, a Multi-Objective Genetic Algorithms (MOGA) will be the EC approach of choice for this problem [1]. In our future work we, hope to proceed along these lines to build MTD systems for SDNs using EC.

## REFERENCES

[1] C. Bacquet, A. N. Zincir-Heywood, and M. I. Heywood. 2011. Genetic optimization and hierarchical clustering applied to encrypted traffic identification. In *2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*. 194–201. DOI: http://dx.doi.org/10.1109/CICYBS.2011.5949391

[2] Ankur Chowdhary, Sandeep Pisharody, and Dijiang Huang. 2016. SDN Based Scalable MTD Solution in Cloud Network. In *Proceedings of the 2016 ACM Workshop on Moving Target Defense (MTD '16)*. ACM, New York, NY, USA, 27–36. DOI: http://dx.doi.org/10.1145/2995272.2995274

[3] HP Corporation. 2013. Ending the Confusion about Software-Defined Networking: A Taxonomy. http://static.itlibrary.nl/downloads/647_HP-6_Whitepaper-Networking.pdf. (2013). Accessed: 2017-01-31.

[4] David J. John, Robert W. Smith, William H. Turkett, Daniel A. Cañas, and Errin W. Fulp. 2014. Evolutionary Based Moving Target Cyber Defense. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO Comp '14)*. ACM, New York, NY, USA, 1261–1268. DOI: http://dx.doi.org/10.1145/2598394.2605437

[5] Douglas C. MacFarland and Craig A. Shue. 2015. The SDN Shuffle: Creating a Moving-Target Defense Using Host-based Software-Defined Networking. In *Proceedings of the Second ACM Workshop on Moving Target Defense (MTD '15)*. ACM, New York, NY, USA, 37–41. DOI: http://dx.doi.org/10.1145/2808475.2808485

[6] Rui Zhuang, Scott A. DeLoach, and Xinming Ou. 2014. Towards a Theory of Moving Target Defense. In *Proceedings of the First ACM Workshop on Moving Target Defense (MTD '14)*. ACM, New York, NY, USA, 31–40. DOI: http://dx.doi.org/10.1145/2663474.2663479