# Characterizing Network-Based Moving Target Defenses

Marc Green, Douglas C. MacFarland, Doran R. Smestad, Craig A. Shue
Dept. of Computer Science, Worcester Polytechnic Institute
Worcester, MA, USA
marcgreen@cs.wpi.edu, dcmacfarland@cs.wpi.edu, doransmestad@cs.wpi.edu,
cshue@cs.wpi.edu

## ABSTRACT

The moving target defense (MTD) strategy allows defenders to limit the effectiveness of attacker reconnaissance and exploitation. Many academic works have created MTDs in different deployment environments. However, network-based MTDs (NMTDs) share key components and properties that determine their effectiveness. In this work, we identify and define seven properties common to NMTDs which are key to ensuring the effectiveness of the approach. We then evaluate four NMTD systems using these properties and found two or more key concerns for each of the systems. This analysis shows that these properties may help guide developers of new NMTD systems by guiding the evaluation of these systems and can be used by others as a rubric to assess the strengths and limitations of each NMTD approach.

## 1. INTRODUCTION

Moving target defenses (MTDs) are designed to provide defenders with an information advantage over attackers: by constantly changing the location of organizational assets, in a method known only to the defenders, information gleaned from attacker reconnaissance quickly becomes stale and inaccurate. This approach raises costs for adversaries to identify and exploit potentially vulnerable systems. Further, when combined with honeypots, which are essentially sinks designed to draw attacker traffic to allow study, the approach can create uncertainty and risk for attackers.

The moving target concept has created a rich body of research, including over 120 academic papers related to the idea [11]. This work can be divided further into strategies for creating moving targets within networks, inside specific hosts, or even within applications themselves (such as the popular ASLR strategy [13]).

When focusing just on network-based MTDs (NMTDs), certain patterns emerge. In this work, we identify the key components involved in these systems and have analyzed their attributes. In doing so, we have identified a set of key properties that each NMTD should meet. These properties can be used as a rubric to assess the quality of a NMTD system and can help NMTD designers ensure that an approach will be robust to attack.

In this work, we make the following contributions:

- **Identification of Common NMTD Components and Properties:** We describe the key components that each NMTD examined has and the properties these systems must guarantee to be effective.
- **Analysis of Four Prior NMTDs:** Using each of the properties we have identified, we assess four previously published NMTD systems, which appear to be exemplars of all identified NMTDs.

In making these contributions, we found that three of the NMTD proposals did not specify or evaluate at least one key property associated with the NMTD. As a result, it is not possible to determine the viability of these proposals based on their publications. By providing these properties and showing how they can be applied to real systems, we hope to help other NMTD creators effectively specify and evaluate their approaches.

## 2. RELATED WORK

Okhravi *et al.* [11] identified over 120 academic papers related to the notion of a moving target defense in computer systems. Their scope is broad with only brief discussion about the types of network-based moving target systems. Our work focuses on network-based MTDs and goes into detail about the properties these systems must consider.

We will evaluate four previously proposed network MTDs in detail. Here, we briefly describe other prominent network MTDs and how they relate to those four systems.

The moving target approach can be used across each of the network layers. Perhaps the most ubiquitous network-based moving target approach is the wireless frequency hopping strategy used to avoid jammers [10]. This approach attempts to evade an adversary that cannot feasibly block all frequencies simultaneously and uses channel hopping to avoid the active blocking attempts. While this strategy uses a different entropy space and identifiers (namely a chosen frequency within the range of available 802.11 frequencies), its considerations are quite similar to the MT6D approach [4] we discuss in detail, which also requires agreement between hosts on hopping patterns.

At the network layer, the DYNAT work by Kewley *et al.* [7] created a NMTD system by modifying the source and destination hosts to translate network addresses in a coordinated way. Atighetchi *et al.* [2] created a system that allows hosts to use tunnels to disguise on-going communication,



*MTD'15 October 12 2015, Denver, CO, USA*



ISBN 978-1-4503-3823-3/15/10…$15.00

DOI: http://dx.doi.org/10.1145/2808475.2808484

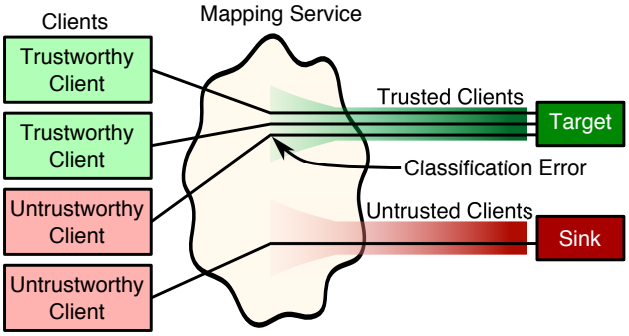


**Figure 1: Overview of components in MTD system**

which requires participation from both end-points. In the NASR approach by Antonatos *et al.* [1], the authors simply changed network addresses on hosts without concern for established connections. They argue that applications will attempt to re-establish the connection automatically, avoiding the need for connection continuity. These approaches are each similar to the MT6D approach, which requires coordination across end-hosts, and can be evaluated similarly.

Port knocking [8] allows a defender to only authorize connections to a server after a special sequence of packets, to specified ports, are received from a client. The approach essentially uses transport layer ports as moving target space for establishing a connection. This approach requires both the client and the server to previously share a key (in this case, used to determine the ports to knock) to establish a connection. Accordingly, the approach can be analyzed similarly to the MT6D approach.

Ge *et al.* [5] propose a cost metric for network-based moving target defense systems to describe how a mapping service could be used to send higher risk users to systems that are particularly well fortified. This formal cost analysis is compatible with our own but is independent of the MTD used.

Each of the NMTDs from related work map to one of the NMTDs that we analyze in depth. This suggests that the analysis and properties we have identified may be common across a large number of NMTDs.

# 3. FUNDAMENTAL NMTD PROPERTIES

There are several components that are common across NMTDs, as depicted in Figure 1. These components include *client* hosts that would like to access a server that is protected by the NMTD system. Some clients are *trustworthy clients* that are not malicious and act in accordance with the destination's goals. Once a client has been granted access to the destination, it is considered a *trusted client*. A *target* is a legitimate server that is protected by the NMTD system while a *sink* is the destination for untrusted targets (which may be an unroutable destination or a heavily monitored honeypot system). The *mapping system* consists of the access control component of the NMTD system and is responsible for discriminating between clients and determining which to authorize. Once it does so, the mapping system may provide *authorization* explicitly (e.g., using a capability) or implicitly (e.g., by manipulating network configurations) to allow the client to reach the target.

A moving target defense system typically manipulates the mapping system and the manifestation of the authorization to achieve its goals. For many systems, the goals achieve three properties: the moving property, the access control property, and the distinguishability property. We now elaborate on each.

## 3.1 Moving Property

By continually altering network information, the NMTD can achieve two goals: 1) force the clients to engage with the mapping system to reach the intended target, which facilitates access control, and 2) limit the window of utility of information that any untrustworthy clients may have gleaned about the network. To be effective at this, the system must meet three additional sub-properties: unpredictability, vastness, and periodicity.

The **unpredictability sub-property** guarantees that a NMTD system must move its targets in a seemingly random manner to clients that lack authorization. Clients should not be able to guess the new destination of any given target unless the client has an active authorization. This property must hold even if the client has an expired or revoked authorization to the target from a previous interaction.

The **vastness sub-property** guarantees that the destination space of the NMTD must be sufficiently large enough so that it is intractable for a client to gain access to a target by exhaustively enumerating all possible destinations. The vastness sub-property can be satisfied by having an extremely large destination space (e.g., using IPv6 addressing) or by being able to detect and mitigate exhaustive enumeration attempts (e.g., proactive blocking of port scanning or IP address scanning hosts). Importantly, any mitigation efforts must be resistant to adversaries controlling botnets in which enumeration attempts can be distributed across seemingly unrelated source hosts.

The **periodicity sub-property** guarantees that targets should be moved with enough regularity that any reconnaissance collected by untrusted clients expires quickly. Defenders may have different goals and strategies which may affect the periodicity requirements and the frequency at which movements should occur.

## 3.2 Access Control Property

The access control property requires a client to reach its target if and only if the client has an active authorization by the mapping system. This property allows the enforcement of policy tailored to authorized and unauthorized systems. However, the property also mandates that authorized parties be able to reach the destination. The property has three sub-properties: uniqueness, availability, and revocability.

The **uniqueness sub-property** guarantees that each client must be individually authorized and that such an authorization cannot be shared with any other client. For a client to access a target, it must successfully have met the precondition of being authorized by the mapping system. This property essentially ensures that the mapping system has explicitly authorized each party, providing detailed situational awareness and control over network traffic.

The **availability sub-property** guarantees that if a client is authorized to contact a given target, the client will be able to successfully reach the target when desired. While availability may ultimately be determined by the capacity of the network and system, a NMTD must ensure that it introduces no new denial-of-service (DoS) vulnerabilities or inherent resource exhaustion limitations.

The **revocability sub-property** allows the mapping sys-

tem to terminate or expire a prior authorization without causing collateral damage. A revocation should not disrupt any other clients or other authorizations from that same client to other authorized targets.

### 3.3 Distinguishability Property

The distinguishability property means a system can separate trustworthy clients from untrustworthy ones. There must be some set of characteristics that trustworthy clients possess and untrustworthy clients do not (or vice versa). The distinguishability property captures the NMTD's ability to use those characteristics in determining which clients should be authorized. Any classification errors will result in either allowing potentially malicious clients to reach a server or denying legitimate clients access to a server.

In some NMTD implementations, the underlying distinguishability property is tied to the adversary's knowledge of the NMTD itself. Since NMTDs are relatively uncommon on the Internet, deployers may distinguish between clients that interact with the mapping system from those that do not (e.g., scanning botnets). Other NMTDs may use identifiers, such as pre-shared keys, to distinguish clients.

## 4. EVALUATING NMTDS

We now classify four existing NMTDs according to the identified properties to highlight the relative strengths and weaknesses of each approach.

### 4.1 DNS Capabilities Approach

In earlier work [15], we proposed a network capabilities system for server defense. In that NMTD, the mapping system was implemented using two distinct components: a DNS server, which provided clients with the destination IP address of a targeted server at that moment, and a NAT mapping device, which granted the client access to the server when a destination mapping was used. The NAT device stored state for each established flow, allowing the DNS server to change addresses and providing a unique response to each new client without disrupting the established flow.

**Moving Property:** Under the system we built, the DNS server provides a unique server IP address to each client with a short (e.g., 5 second or less) time-to-live (TTL) on the DNS record. The NAT device uses this response to create a corresponding available window on that IP address for the client to connect. If the client connects during this time, the NAT device establishes a mapping to allow the client to reach the target. Otherwise, the NAT device's default rule sends the client to the sink.

Since the DNS server used a pseudo-random number generator to pick IP addresses for each new client, this approach fulfills the requirements of the unpredictability and periodicity sub-properties.

With IPv6, the vastness sub-property is easily met: with a standard 64-bit prefix assignment, even with a scan rate of one billion hosts per second, it would take an adversary over 500 years to enumerate the search space and find an open IP address, if any. In an IPv4 network, the approach loses this advantage. However, defenders may restrict the source addresses that may use each opening. In other prior work [14], we proposed linking DNS resolvers with their clients to minimize the number of clients that could share the capability. Most resolvers and clients are located in the same autonomous system, allowing a defender to only allow clients from the same ASN as the resolver to use the capability provided by the DNS server[1]. Since the largest ISP controls only 2% of the IPv4 address space, 98% or more of Internet hosts would be unable to successfully guess the valid IP address simply because they would not be from the appropriate source network.

**Access Control Property:** Due to DNS caching, it is possible for clients to share a returned address. This technically allows two clients to share a capability; however, it is challenging for an adversary to do so. Without any other information, the adversary must use a client from the same AS, guess the correct IP address from the IP range at the destination, and issue a connection request during the short TTL availability window. Accordingly, we consider the uniqueness sub-property met, with these caveats.

Since NAT devices must already track flows, the approach does not hinder availability.

To revoke a network capability, the mapping system can simply remove any NAT mappings between the client and the target. This will automatically redirect the client's flows from the target to the sink without affecting any other clients or the client's connections to any other servers. Even if multiple clients happen to share a capability initially, due to DNS cache effects, the establish network flows table stores a separate flow entry for each client. The mapping system can then revoke access individually.

**Distinguishability Property:** Compromised machines engaged in scanning [17], such as for bots that attempt SSH brute-force login attempts, do not perform DNS lookups. Even Web crawlers, which often perform DNS pre-fetching, may not retrieve the page within the specified TTL, causing access to be denied. However, the approach cannot distinguish between legitimate clients and automated attackers that immediately request and use the DNS capability.

Our prior work [14] noted that passive techniques could be used to distinguish ISP-provided DNS resolvers from those running on client systems. Future work could integrate these mechanisms into the DNS capabilities system itself to better distinguish legitimate users from adversaries. In particular, attackers may avoid using the ISP DNS infrastructure associated with each compromised system to avoid detection when performing large-scale scanning.

### 4.2 OpenFlow Mutation

Shortly after the DNS capabilities approach, Jafarian *et al.* [6] proposed a similar moving targets approach using IP address randomization in a LAN using the OpenFlow protocol. In the OpenFlow protocol, network switches can be configured as high-speed network caches. When the switch receives a packet and does not know how to forward the packet, it can ask for instructions (called *elevation*) from a specific machine, called the OpenFlow controller.

The OpenFlow mutation approach uses this elevation mechanism to alter DNS records and to change packet addresses in flight. When a client performs a DNS lookup, the DNS server will provide the client with a virtual IP address for a target. When the client attempts to access the target,

[1]Remote DNS resolvers, such as OpenDNS and Google, can provide subnet information for the client making the request. Consult http://www.afasterinternet.com/howitworks.htm for details.

**Table 1: Summary of NMTDs by Property**

| Properties | | DNS Capabilities | OpenFlow Mutation | MT6D | Simulated MTD |
|---|---|---|---|---|---|
| Moving | Unpredictability | Yes | Yes | Yes | Yes |
| | Vastness | Yes | Yes | Yes | Unaddressed |
| | Periodicity | Yes | Yes | Yes | Unaddressed |
| Access Control | Uniqueness | Usually | Yes | Yes | Unaddressed |
| | Availability | Yes | Unaddressed | Unaddressed | Unaddressed |
| | Revocability | Yes | Yes | Yes | Unaddressed |
| Distinguishability | | Partial | Partial | Yes, requires client modifications | Yes, requires client modifications |

the packet is forwarded using the virtual IP address until it reaches the destination switch, at which point it is translated to the destination host's real IP address. The network switches essentially act as NAT devices, performing translations between virtual and real IP addresses.

The OpenFlow controller essentially serves as the mapping system. The controller updates the DNS server with different virtual IP addresses for each system and orders the OpenFlow switches to install NAT rules to hide the identities of both hosts. Unfortunately, since the controller must have control of the source and destination switch, the approach is most suited to a LAN.

**Moving Property:** The OpenFlow mutation approach specifies that virtual addresses can be selected at random or in a weighted random fashion, using only information known to the defender, for each connection. It thus meets the unpredictability and periodicity sub-properties.

The approach easily provides vastness when using IPv6 addressing. In IPv4, an adversary may enumerate the possible combinations, but this scanning behavior can be easily detected by the OpenFlow controller. The controller can meet the vastness sub-property by blocking the host.

**Access Control Property:** The OpenFlow mutation approach meets the uniqueness sub-property. The OpenFlow controller can distinguish the requests from clients without an intermediary DNS resolver acting as a proxy. Likewise, the revocability sub-property is met since the OpenFlow controller can revoke authorization by replacing the flow translations at the network switches with drop rules to terminate access between the hosts.

The availability sub-property is met if the OpenFlow controller and switches can meet the demands of traffic elevation and forwarding. These are concerns about scaling fine-grain flows in OpenFlow networks [3, 9] and these scaling features car be used by adversaries to create network denial-of-service conditions [12]. These constraints were not explicitly discussed in the work, so it is unclear if the availability sub-property is met.

**Distinguishability Property:** As with the DNS capabilities approach, the OpenFlow mutation approach can only distinguish between clients that use the DNS process and those that perform scanning.

## 4.3 MT6D

Dunlop *et al.* [4] created the MT6D "Moving Target IPv6 Defense" system in which the client and target share a symmetric key out-of-band and use these keys to determine the IPv6 addresses the hosts will use. To construct their IPv6 addresses, the hosts construct a hash using the shared key, a value derived from the host's MAC address, and a timestamp. The approach extracts 64 bits from the hash output to encode the lower 64 bits of the host's IPv6 address. To provide uniform communication between the host applications, the operating system on each host uses a tunneling approach and rotates the addresses of the tunnel end-points.

The mapping service is implemented on the hosts themselves: the mapping service is divided into the hashing operation and the kernel addressing and tunneling systems, which update the addresses.

The security goals of this approach are different from the DNS capabilities and OpenFlow mutation approaches. Given that MT6D modifies both end hosts, uses a pre-shared symmetric key, and tunnels network traffic, MT6D could use an established VPN protocol, such as IPSec, to achieve the access control and distinguishability properties.

**Moving Property:** MT6D's use of a keyed hash function meets the unpredictability sub-property. MT6D meets the vastness sub-property by using moving among the lower 64 bits of an IPv6 address. Finally, MT6D meets the periodicity sub-property by allowing extremely rapid transition among addresses. In evaluating the approach, the authors described an experiment using a 10 second interval.

**Access Control Property:** The MT6D authors do not explicitly state that the shared key between the client and the target are unique. If the shared key is not unique, MT6D fails to meet the uniqueness and revocability sub-properties, since the target must authorize hosts in groups. However, if the MT6D approach uses unique keys, MT6D can provide these sub-properties.

The authors do not show that the availability sub-property is met. In particular, the authors did not evaluate the impact of address rotation on the network infrastructure near the hosts. The approach could exhaust network state at the routers, but this factor is not evaluated.

**Distinguishability Property:** Since MT6D requires a unique symmetric key at both hosts, the MT6D approach can successfully discriminate against unauthorized users that lack such symmetric keys.

## 4.4 Simulation-based MTD

Zhuang *et al.* [16] proposed using simulations to study moving target defenses. The authors modified the clients and targets with special MTD software that orchestrates the movements of hosts. Each client and target ran its own mapping service. The actual approach to connect from a client to a target is presented generically; the authors discuss both a proxying approach similar to MT6D and an alternative using a special API for communication.

While the authors specifically designed an abstract and generic concept for their MTD, they did not provide guidance on meeting the essential NMTD properties.

**Moving Property:** The simulation approach requires "chaotic" movements that appear to meet the unpredictability sub-property. However, it is unclear whether the approach meets the vastness or the periodicity sub-properties since the work does not explicitly discuss these properties or specify constraints.

**Access Control Property:** The uniqueness and revocability sub-properties are not clear in the approach. It may be possible to ensure uniqueness by avoiding duplicated roles across clients and targets. If so, revocability can also be guaranteed since the resource mapping system can eliminate prior authorization by rotating resources and not providing a new mapping according to the roles. The availability sub-property is not explicitly addressed by the approach.

**Distinguishability Property:** As with the MT6D approach, this system can distinguish between authorized systems, which have been pre-configured with the MTD software, from systems lacking such configuration.

# 5. CONCLUSION

In this work, we described three key properties for network moving target defense systems: the moving property (including the predicability of the movement, the vastness of the movement space, and the periodicity of the movements), the access control property (including uniqueness, availability, and revocability), and the distinguishability property (to discriminate between legitimate and malicious traffic). Upon articulating and motivating these properties, we evaluated four network MTDs.

Our evaluation of the NMTDs found that three of them had not properly specified and evaluated at least one property that is essential to determining whether the approach is viable in practice. With this work, we hope to help future NMTD designers recognize the key elements they must consider when designing and evaluating a NMTD. Further, we hope our properties will serve as a basic rubric that can be used by others to assess the key strengths and limitations of network-based moving target systems.

# 6. REFERENCES


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

[2] M. Atighetchi, P. Pal, F. Webber, and C. Jones. Adaptive use of network-centric mechanisms in cyber-defense. In *Symposium on Object-Oriented Real-Time Distributed Computing*, pages 183–192. IEEE, 2003.

[3] A. R. Curtis, J. C. Mogul, J. Tourrilhes, P. Yalagandula, P. Sharma, and S. Banerjee. Devoflow: Scaling flow management for high-performance networks. In *Proceedings of the ACM SIGCOMM 2011 Conference*, SIGCOMM '11, pages 254–265, New York, NY, USA, 2011. ACM.

[4] M. Dunlop, S. Groat, W. Urbanski, R. Marchany, and J. Tront. MT6D: A moving target IPv6 defense. In *Military Communications Conference*, pages 1321–1326. IEEE, 2011.

[5] L. Ge, W. Yu, D. Shen, G. Chen, K. Pham, E. Blasch, and C. Lu. Toward effectiveness and agility of network security situational awareness using moving target defense (MTD). In *SPIE Defense+ Security*, pages 90850Q–90850Q. International Society for Optics and Photonics, 2014.

[6] J. H. Jafarian, E. Al-Shaer, and Q. Duan. Openflow random host mutation: Transparent moving target defense using software defined networking random host mutation: transparent moving target defense using software defined networking. In *Hot Topics in Software Defined Networks*, pages 127–132. ACM, 2012.

[7] D. Kewley, R. Fink, J. Lowry, and M. Dean. Dynamic approaches to thwart adversary intelligence gathering. In *DARPA Information Survivability Conference and Exposition*, volume 1, pages 176–185. IEEE, 2001.

[8] M. Krzywinski. Port knocking from the inside out. *SysAdmin Magazine*, 12(6):12–17, 2003.

[9] A. Lazaris, D. Tahara, X. Huang, E. Li, A. Voellmy, Y. R. Yang, and M. Yu. Tango: Simplifying SDN control with automatic switch property inference, abstraction, and optimization control with automatic switch property inference, abstraction, and optimization. In *ACM International on Conference on Emerging Networking Experiments and Technologies*, pages 199–212, New York, NY, USA, 2014. ACM.

[10] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein. Using channel hopping to increase 802.11 resilience to jamming attacks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 2526–2530. IEEE, 2007.

[11] H. Okhravi, T. Hobson, D. Bigelow, and W. Streilein. Finding focus in the blur of moving-target techniques. *Security & Privacy, IEEE*, 12(2):16–26, 2014.

[12] S. Scott-Hayward, G. O'Callaghan, and S. Sezer. SDN security: A survey. In *IEEE SDN for Future Networks and Services*, pages 1–7, Nov 2013.

[13] H. Shacham, M. Page, B. Pfaff, E.-J. Goh, N. Modadugu, and D. Boneh. On the effectiveness of address-space randomization. In *ACM Conference on Computer and Communications Security*, pages 298–307. ACM, 2004.

[14] C. Shue and A. Kalafut. Resolvers revealed: Characterizing DNS resolvers and their clients. *ACM Transactions on Internet Technology (TOIT)*, 12(4), July 2013.

[15] C. A. Shue, A. J. Kalafut, M. Allman, and C. R. Taylor. On building inexpensive network capabilities. *ACM SIGCOMM Computer Communication Review*, April 2012.

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

[17] C. C. Zou, D. Towsley, and W. Gong. On the performance of Internet worm scanning strategies. *Performance Evaluation*, 63(7):700–723, 2006.