Using an IPv6 Moving Target Defense to Protect the Smart Grid

Stephen Groat, *Member, IEEE,* Matthew Dunlop, *Member, IEEE,* William Urbanksi, *Member, IEEE,*
Randy Marchany, *Member, IEEE,* and Joseph Tront, *Senior Member, IEEE*

***Abstract*—As advanced Internet Protocol (IP)-based communication systems are proposed for the Smart Grid, security solutions must be developed which address not only the security of the communications, but also the security of the communicating systems. To support the large number of hosts required for the Smart Grid on an IP network, the new Internet Protocol version 6 (IPv6) must be leveraged. Unfortunately, IPv6 inherits the majority of Internet Protocol version 4 (IPv4) vulnerabilities as well as adds new address-based exploits. The embedded systems necessary for Smart Grid deployments using IP communications will be especially vulnerable to attacks due to their limited system resources. A moving target defense not only secures the communications between peers, but also prevents the peers from being located for attack. Implementing security at the network layer mitigates most IP-specific exploits and allows for solutions to be integrated with minimal impact to existing Smart Grid systems, thus reducing costs and increasing reliability. By using a network layer moving target defense, hosts cannot be located for exploitation and secure connectivity is maintained with trusted peers. A robust Smart Grid network must be capable of proactive defense so that components are not consumed with defending incoming attacks. A solution which implements a proactive network layer defense called the Moving Target IPv6 Defense (MT6D) is offered as a potential solution for secure Smart Grid communications.**

I. Introduction

The Smart Grid is designed to provide consumers with reliable, efficient, and safe electric energy. By incorporating advanced power system controls and communications between consumers and the power infrastructure, consumers will have more visibility and control over their energy usage. Likewise, electric providers will be able to actively adapt to changing loads and compensate for failures [1]. The most likely communication system to facilitate this information exchange is based on the well-established Internet Protocol (IP). By using IP-based communications, the Smart Grid will be able to leverage the existing Internet infrastructure, avoiding the unnecessary complexity and expense of creating a separate communications infrastructure. In order to support the immense number of extra Internet connected devices expected with the Smart Grid deployment, a new IP must be adopted.

The Internet Protocol version 4 (IPv4), the current standard for Internet communications, is running out of available addresses. The Internet Assigned Numbers Authority (IANA) assigned its remaining five IPv4 address blocks on 3 February 2011 [2]. The Internet Protocol version 6 (IPv6) provides a solution to the address depletion problem by allocating 128 bits for addresses verses 32 bits available in IPv4. The size of the IPv6 address space equates to approximately $5 \cdot 10^{28}$ addresses for every one of the 6.8 billion people [3] in the world. This immense address space is more than enough to support global addressability in the Smart Grid for the foreseeable future. While the problem of limited address space has been solved, the vulnerabilities associated with networked communications have not. Current network-level attacks and vulnerabilities associated with computer systems make all IP-based Smart Grid devices vulnerable to attack.

As new systems and networks are built to deploy the Smart Grid, the unique and potentially open network architecture of the Smart Grid's communications systems must designed with security in mind. Adding security to the Smart Grid is made more challenging due to the limited resources of the embedded meters, sensors, and other devices. Using IP-based communications in the grid brings numerous IP-based vulnerabilities. Attackers can exploit these vulnerabilities to manipulate either the consumers' devices or the power infrastructure itself. Both application-layer, or system specific attacks, and network layer attacks, such as denial of service, are possible. As a result, many researchers are investigating ways to secure the Smart Grid [4], [5]. Most current security systems are static, allowing attackers an unlimited number of attack attempts. Given time, a committed attacker is likely to find a way through the various layers of static defenses. A moving target defense prevents the targeting of nodes by proactively changing characteristics about nodes before a threat is detected. These changes give nodes the appearance of continuously moving. As a result, attackers are forced to use time and resources to find and reacquire targets as they move.

Since most network attacks target specific addresses, an effective defense moves the nodes' network addresses. This is not feasible in the IPv4 address space because of its dense population and the ease of locating hosts through exhaustive scanning [6]. In IPv6, however, individual subnets are vastly larger and infeasible to scan using current technology. In fact, a *single* IPv6 subnet is over four billion times larger that the entire IPv4 address space and takes $8.77 \cdot 10^{10}$ years to scan exhaustively [6]. Therefore, an effective network-layer moving target solution that proactively defends systems should to be based on IPv6.

The remainder of this paper is organized as follows. Section II presents necessary background information pertaining

S. Groat, M. Dunlop, and J. Tront are with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, 24061 USA e-mail: {sgroat, dunlop, jgtront}@vt.edu.

W. Urbanski and R. Marchany are with Information Technology Security Office, Virginia Tech, Blacksburg, VA 24061 USA e-mail: {urbanski, marchany}@vt.edu.

978-1-4577-2159-5/12/$31.00 ©2011 IEEE

to IPv6 and the Smart Grid. Other pertinent research related to securing IPv6 and the Smart Grid is discussed in Section III. The problems faced by Smart Grid deployment and embedded systems are described in Section IV, while Section V presents a possible security solution. The design of this security solution is detailed in Section VI. Section VIII discusses some limitation of applying this solution to the Smart Grid. Section IX addresses future work with regard to securing the Smart Grid. Concluding remarks are provided in Section X.

II. Background

The large number of embedded systems that are necessary components of the Smart Grid surpasses the address capabilities of IPv4. It will be necessary to build the Smart Grid over IPv6 to allow for globally addressable sensors and meters to communicate over the existing Internet infrastructure. As already mentioned, IPv6 provides a vastly larger address space. Whereas an IPv4 address, consisting of 32 bits, provides approximately 4.2 billion possible address combinations, the IPv6 address is expanded to 128 bits. This new address size allows for 2^{128} or $3.4 \cdot 10^{38}$ possible addresses. The immense size of the IPv6 address space requires new address management strategies. Unfortunately, the strategies proposed have potential vulnerabilities associated with them. In addition to the addressing vulnerabilities, some of the design specifications necessary for embedded systems are easily exploitable.

A. Stateless IPv6 Addressing

The large size of the IPv6 address space requires a new network address configuration architecture to simplify network administration. For this reason, IPv6 combines a Neighbor Discovery Protocol (NDP) [7] with StateLess Address Auto-Configuration (SLAAC) to allow for nodes to self-determine their IP addresses. Designed as a replacement for the Address Resolution Protocol (ARP), NDP facilitates nodes within a particular subnet learning of other nodes on the link using Internet Control Message Protocol version 6 (ICMPv6) messages. Once a NDP message is received, the node uses the network portion of the address to configure the first 64 bits of its IPv6 address. For the last 64 bits, the node automatically configures an address, designated as the interface identifier (IID) of the address. The final step combines the 64-bit network address with the 64-bit host address to form a complete 128-bit IPv6 address.

Due to the current accepted definition of SLAAC on most operating systems, the IID of a node's IPv6 address is deterministic across networks. For the last 64 bits, the node automatically configures an address based upon the Media Access Control (MAC) address of its network interface. While different operating systems configure IPv6 addresses differently, no current operating system implementations of IPv6 stateless addressing dynamically obscure the IID of all IPv6 addresses on the system. Not dynamically obscuring the IID for all of the IPv6 addresses associated with a system threatens a system's privacy, anonymity, and security. The static IID currently implemented in major operating systems can be linked to a particular node and targeted, even if the node changes networks.

B. Stateful IPv6 Addressing

The inherent privacy and security flaws in SLAAC will lead Smart Grid designers to consider the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) as a stateful addressing alternative. DHCPv6 allows administrators more control over address distribution than SLAAC. It provides many of the same features as the Dynamic Host Configuration Protocol (DHCP) implemented in IPv4, including logging and auditing capabilities. To provide more precise auditing and configuration, DHCPv6 deploys a DHCP Unique Identifier (DUID). The DUID is used to identify unique hosts to the server. While DHCPv6 is a stateful protocol which can be configured to change nodes' addresses frequently, the static DUID is exposed in many common DHCPv6 messages and allows nodes to be tracked.

The DUID allows for correlation of users' addresses over multiple sessions, creating similar privacy and monitoring concerns as SLAAC. While the RFC states that the DUID is optionally static, current operating system (OS) client implementations use a permanent DUID. Since the DUID for a client persists between sessions and networks, nodes can be geotemporally tracked and have their traffic correlated. Typically, attackers must be on the link-local network to receive the SOLICIT and ADVERTISE messages containing DUIDs. They can also remotely monitor and track clients by planting modified relays at targeted sites to forward multicast DHCPv6 messages for analysis. Once the DUID contained in the DHCPv6 message is obtained, attackers can sniff the network for the DHCPv6 response or query the DHCPv6 server for the nodes' addresses. Since stateful addressing has the same privacy concerns as SLAAC, IPv6 address assignment alternatives must be improved before IPv6 deployment weakens privacy and security for Smart Grid systems and consumers.

C. Embedded Systems Vulnerabilities

Networked embedded systems share similar vulnerabilities as all network computer systems, but their limited resources open them to unique classes of attack. Currently, system vulnerabilities are often exposed in the network layer and application layer. Denial-of-Service (DoS) attacks tend to disrupt network communications while application layer exploits attempts to manipulate and exploit OS and application features. Systems are also vulnerable to physical exploits, such as side-channel attacks. Current security procedures can address these vulnerabilities for embedded system in the Smart Grid. Yet, how Smart Grid embedded systems will be integrated into the system's infrastructure and control mission-critical processes creates new classes of attacks focusing on unique characteristics of embedded systems.

Embedded systems are optimized to operate in real-time with minimal power consumption. The design constraints of the limited resources of the system often amplify vulnerabilities. As such, they are vulnerable to unique classes of both time-based and energy-based attacks [8]. The goal of time-based attacks are to force targeted system to miss scheduled deadlines and therefore either miss critical tasks or become out-of sync. Time-based attacks can be accomplished through DoS attacks. Since embedded systems are limited in their computational power, an attacker can overwhelm an embedded system without considerable effort. Energy-based attacks are designed to force embedded systems that rely on battery power to expend large amounts of computational power, thus prematurely draining their batteries. There are numerous ways that monitoring systems can be made to expend energy. These range from continuously sending bogus reports that the embedded system have to process to repeatedly turning utilities on and off that embedded systems are responsible for managing [8].

III. Related Work

Numerous other researchers have investigated methods of securing the Smart Grid against attack. Most of this research is focused on static protection schemes. Surprisingly, no research was found related to moving target defenses. The closest research to this topic focuses on self-healing capabilities within the Smart Grid. This section presents some of the research that attempts to provide security through self-healing capabilities. First, some related research is described relating to protecting IPv6 addresses and embedded systems.

A. IPv6 Address Protection

Some research investigates IPv6 address protection. This research can be classified as stateless or stateful. Stateless address protection involves obscuring the IPv6 address. There are two static methods and one dynamic method for accomplishing this. Stateful address protection involves obscuring the DUID. Currently, only one method is proposed to accomplish this.

Two static address obscuration techniques have been proposed to achieve anonymity in SLAAC. The two proposals are privacy extensions [9] and Cryptographically Generated Addresses (CGAs) [10]. Privacy extensions were designed specifically with the intent of obscuring IPv6 addresses. CGAs were designed to securely associate IPv6 addresses with public keys for use with the SEcure Neighbor Discovery (SEND) protocol. This association also hides the address of the packet source. Neither of these schemes dynamically obscure addresses. Once an address is assigned, it remains constant, minimally, until the network session is terminated. A third party monitoring the connection can accomplish both address tracking and traffic correlation. These techniques also only obscure the source address. Traffic correlation can still occur by monitoring source and destination address pairs. A dynamic technique was proposed by Dunlop et al. [11]. This technique and how it applies to the Smart Grid is described in Section VI.

As discussed in Section II, DHCPv6 DUIDs can provide an attacker with a means to identify and target a specific node. Groat et al. [12] propose a method of dynamically obscuring DUIDs that leverages self-generated history values similar to those created in privacy extensions [9]. By obscuring DUIDs, attackers are unable to tie a specific DUID to a host. The weakness of this approach is that the current design of DUID tables within DHCPv6 do not support a large number of stored DUIDs. Therefore, performance is impacted by incorporating dynamic DUIDs. For this approach to be practical, DHCPv6 DUID tables would need to be redesigned.

B. Security in Embedded Systems

The vulnerabilities associated with embedded devices are well-documented [13], [14]. Aside from traditional communication system vulnerabilities, many embedded systems are faced with resource consumption attacks due to limited computational capability and battery power. Resource consumption attacks are aimed at denying service to embedded systems. These type of attacks can be devastating to Smart grid components, particularly those that are unmanaged in remote locations. Many researchers have proposed solutions to these vulnerabilities. For example, Jacoby et al. [15] proposed a technique that monitored irregularities in battery power output. These irregularities can be used to signal alerts of potential attacks on embedded systems. Although techniques such as this are useful for detecting DoS attacks against embedded systems, they do not prevent them from occurring. A dynamic addressing approach can prevent targeted attacks against Smart Grid components. In the improbable event that an attacker locates a Smart Grid component, dynamic addressing limits the scope of the attack to the duration between address rotations.

C. Self-Healing Smart Grid Defenses

Self-healing capabilities of Smart Grid systems do not prevent attacks, but rather keep the power grid operational in the event of accidental or purposeful failure of Smart Grid components. In some cases, self-healing is achieved through redundancy [16]. In other cases, individual components can act as independent agents either dividing the grid into “islands” or reorganizing to re-establish connections to healthy portions of the grid [17]. Self-healing capabilities are critical, particularly in the event of physical system compromises or failures.

A key component of self-healing systems is post-fault or post-attack recovery. The disadvantage of this approach is that a fault or attack has already occurred. A successful attack means, at the very least, loss of revenue. For optimal reliability of the Smart Grid, a pre-attack posture should be assumed. A moving target defense solution achieves this. If Smart Grid components are statistically infeasible to locate on the network, purposeful compromise is prevented.

IV. Problem

IPv6 is essential to the successful deployment of the Smart Grid. The limited address space in IPv4 requires either Network Address Translation (NAT) or other address translation technology to increase the number of available addresses. While these technologies seem attractive and can effectively enlarge the current IPv4 address space, the configuration and management problems created by using NAT make the technology difficult to scale and cost prohibitive. Alternatively, the large address space and global connectivity offered by IPv6 will allow simple configuration and scalability for the Smart Grid. Yet, the global connectivity created by IPv6 will require security for all systems in the Smart Grid network, especially the embedded meters and sensors.

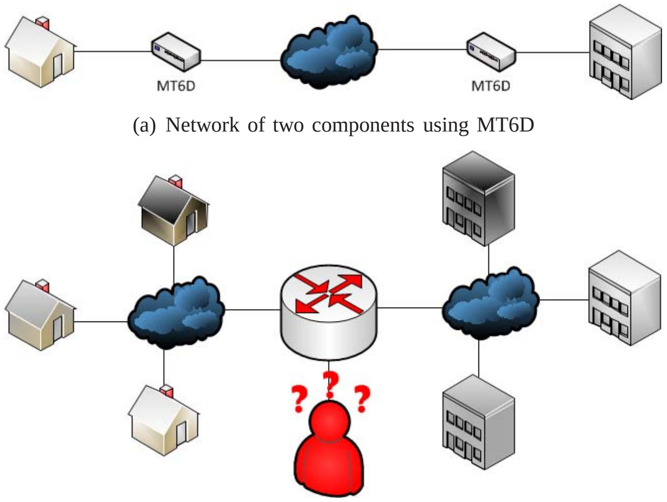
Embedded systems are particularly vulnerable to attack. The limited resources on the embedded systems lead to vulnerabilities such as buffer overflows, allowing for an attacker to gain control of the system and cause increased billing or falsify system load on the network. A simpler class of network layer attacks are those aimed at denying service. A DoS attack against a Smart Grid system may cause service interruptions or system failures if critical communications, such as power load, cannot be communicated. Correlating network traffic sent between Smart Grid components is trivial and can have devastating effects if static addresses are used. An attacker looking to monitor users' consumption patterns may be able to gain valuable information from this information, such as a good time to burglarize a home or business. These vulnerabilities allow for the Smart Grid system to be hijacked, shutdown, or monitored with potentially devastating consequences. In a power failure, the information communicated by Smart Grid meters is even more essential than in times of normal system operation. Power consumption attacks against Smart Grid meters could disrupt this communication, possibly preventing communications in times of power failure. A successful Smart Grid deployment must address security, privacy, and authentication of all traffic on the network.

Although securing the Smart Grid is critical, attempting to add traditional security mechanisms in the development stages will most likely reduce reliability. While the programmers focusing on Smart Grid applications may be experts in power management, safety, and system reliability, forcing them to address system and network security in their applications will likely degrade the overall product. Embedded Smart Grid systems need a network layer defense solution which can holistically protect the system and be added to existing embedded systems without impacting communications or operations. Systems must be protected against targeted network attacks. Also, all traffic must be encrypted and authenticated to protect consumers and their information.

V. Solution

Protecting against targeted network attacks must be the highest priority of any network. To address the vulnerabilities associated with embedded networked systems, proactive defense measures that use minimal system resources, such as a moving target defense, protect a system from being exploited. A moving target defense also protects systems from simple, network level exploits, such as DoS attacks. By eliminating the address as an attack vector, a moving target defense forces attackers to develop new procedures to find and exploit systems. Network level security solutions are also simpler to integrate into existing products since they only modify traffic once it has left the control of the system.

Using a network-level moving target solution adds anonymity to Smart Grid consumers, thus protecting them from power usage exploitation. Power usage can reveal sensitive information about consumers. Businesses could be exploited by analyzing power usage to determine when important meetings are taking place or when labs are conducting experimentation (e.g., above average power usage in a conference room or lab). Smart Grid components can even be exploited to cut power during important conferences or experiments.



(a) Network of two components using MT6D

(b) A third party's view of two Smart Grid components using MT6D; the components appear and disappear on seemingly random addresses

Fig. 1. The actual configuration of two communicating components versus the perceived configuration by a third party

A moving target solution can also provide anonymity for homeowners. Many homeowners leave lights or a radio on to give the appearance of a home being occupied. If detailed information about consumers' power consumption was available, attackers could detect abnormal changes in power consumption in effort to determine the best time to burglarize a home. Changing network addresses makes it difficult for attackers to correlate traffic with hosts. Since network addresses continually change, it is difficult to correlate network traffic to the address of a specific consumer. The additional use of encryption, makes an attacker's already difficult task statistically infeasible.

While traffic must remain anonymous to protect users' identities, Smart Grid communications must also be authenticated to verify users' communications. Without authentication, systems could have communications sent on their behalf without their consent. For example, false requests for updated billing could be sent on a consumers behalf or utilities can send false usage reports. Current traffic authentication systems require for the authenticator's identity be exposed. If authenticated communications are both "moving" and encrypted, attackers cannot tie authenticated traffic to an identity.

VI. Design

Virginia Tech has developed a moving target defense for IPv6 that adds privacy, anonymity, and security to Smart Grid components without impacting communications or operations. The Moving Target IPv6 Defense (MT6D) [11] continually rotates through dynamically obscured network addresses while maintaining existing connections. As discussed in Section IV, static addresses are easy targets for address tracking and network attacks. MT6D prevents attackers from targeting specific addresses by dynamically rotating network and transport layer addresses without impacting preexisting sessions. The dynamic addresses are not linked to specific components, requiring attackers to scan the subnet for targets. The immense address space of IPv6 provides an environment so large that an efficient search is infeasible [6]. In the unlikely event that attackers locate a target, the damage they can inflict is limited to the interval between address rotations; reacquiring the target is infeasible. Fig. 1(a) depicts the actual network configuration of two Smart Grid components while Fig. 1(b) illustrates what two communicating devices look like to a third party.

MT6D modifies the network and transport layer addresses of the sender and receiver nondeterministically. It is capable of dynamically changing these addresses to hide identifiable information about a host, effectively obscuring communicating hosts from any third-party observer. A key feature of MT6D is that this obscuration can be made mid-session between two hosts without causing the additional overhead of connection reestablishment or breakdown. Changing addresses mid-session protects communicating hosts from an attacker being able to collect all packets from a particular session for the purpose of traffic correlation.

MT6D IIDs are computed using three components obscured by a function. The first component is a value specific to an individual host (e.g. a MAC address). The second component is a secret (e.g. symmetric key) shared by the sender and receiver. The third component is a changing value known by both parties (e.g. time). The only one of these three values that must be kept secret is the shared secret. The function results in a 64-bit output used as the MT6D IID and has the form:

$$IID'_{x(i)} = f\{IV_x * S * CV_i\}_{64}$$

where $IID'_{x(i)}$ represents the obscured IID for host x at a particular instance i, IV_x represents a value specific to the individual host x, S represents the shared secret, and CV_i represents the changing value at instance i. The three components are combined using an operation denoted by $*$. The 64-bit function result is denoted by $f\{\cdot\}_{64}$.

In our implementation, each packet is encapsulated in Unreliable Datagram Protocol (UDP) to prevent Transmission Control Protocol (TCP) connection establishment and termination from occurring every time a MT6D address rotates. Encapsulating packets as UDP has a minimal effect on the transport layer protocol of the original packet. Since transport layer protocols are end-to-end, decapsulation will occur before the host processes the original packet. A session using TCP will still exchange all required TCP-related information. This information will simply be wrapped in a MT6D UDP packet. Additionally, any lost packets that were originally TCP will be retransmitted after retransmission timeout occurs.

MT6D provides the option of encrypting each original packet before appending it with the MT6D header. By encrypting the original packet, a third party is unable to glean any useful information. For example, if the original packet is sent using TCP, the header gets encrypted so that a third party cannot attempt to correlate network traffic using the TCP sequence numbers. Additionally, the nature of the network traffic is also kept private through encryption.

Symmetric keys used in the encryption and address creation of MT6D limit access to communicating host pairs on the Smart Grid, creating security through segregation. Only hosts that have established an MT6D connection can communicate with each other since hosts' addresses will mutating non-deterministically. This network segregation helps to prevent system breaches or exploits from penetrating the entire system, limiting their impact.

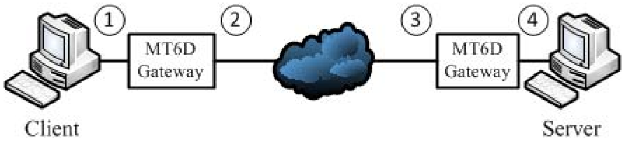


Fig. 2. MT6D testbed. The number 1-4 indicate the network measurement points.

The architecture of a MT6D device mimics a network bridge. Outbound packets are sent to an encapsulator that constructs a MT6D packet. The MT6D packet contains the entire original packet excluding original addresses. When a MT6D packet arrives at its destination, the packet enters a decapsulator which restores the packet to its original form. The design of MT6D facilitates implementation either embedded directly on components or as stand-alone gateway devices. The latter implementation does not require component modification and offloads computation from Smart Grid components.

VII. Testing and Results

A functional prototype of MT6D on the production IPv6 network at Virginia Tech has validated the concept. The initial prototype was designed as a proof of concept to demonstrates the feasibility of MT6D and its ability to send traffic and maintain sessions and state between address rotations. Virginia Tech has a fully functional IPv6 network, providing globally unique addresses through SLAAC to every wireless and wired node on the network. The production network provides us with results that account for the effects of actual network traffic on MT6D packets. The test network is illustrated in Fig. 2. Network traffic between the client and server was routed through the core network, which routes traffic for over 30,000 nodes.

The prototype software was installed on a GuruPlug Server containing a 1.2 GHz ARM CPU with 512MB DDR2 800MHz RAM running 32-bit Debian Linux. The GuruPlug also contains two Gigabit Ethernet Network Interface Controllers (NICs), which were used to separate the trusted and untrusted networks. The configuration used for testing was the gateway implementation discussed in Section VI. This configuration supported isolation of the MT6D prototype from the client/server. It also allowed us to verify the MT6D prototype's ability to pass and accept network traffic from different operating systems.

Our test scenario was primarily aimed at demonstrating the functionality of MT6D. To that end, we tested MT6D's ability to successfully pass different kinds of traffic. For ease of traffic analysis, we set a fixed address rotation interval of 10 seconds. To measure basic functionality, we sent 1000 ping packets from the client to the server at a rate of one packet

TABLE I
ICMPV6 ECHO REQUESTS AND REPLIES OBSERVED AT EACH MEASUREMENT POINT USING MT6D

		Measurement Points						
		1		**2**		**3**		**4**
Standard Ping (1000 packets)	Requests Sent	1000	→	990	→	990	→	980
	Replies Received	979	←	980	←	980	←	980
Ping Flood (10,000 packets)	Requests Sent	10,000	→	9918	→	9918	→	9918
	Replies Received	9741	←	9756	←	9756	←	9918

per second. To test MT6D under a high traffic volume of connectionless traffic, we sent a 10,000-packet ping flood from the client to the server. To test how MT6D handles connection-oriented traffic we had the client use the Hypertext Transfer Protocol (HTTP) over TCP to download files ranging from 1KB to 500MB from the server. We used these tests to gain an understanding of packet loss and latency.

A. *Packet Loss*

Packet loss was easiest to measure when passing connectionless traffic. To measure this we sent 1000 pings from the client to the server at one ping per second. The packet loss was 2.1%. We then sent a 10,000 packet ping flood to the server and observed 2.6% packet loss. Table I shows the number of packets lost at each measurement point depicted in Fig. 2 as well as the direction of packet flow. We attribute the majority of packet loss to packets sent at the address rotation time. When a new address is added to the NIC, the adapter is temporarily disabled while the OS is configured to accept the new address. While this configuration occurs, all the packets sent through MT6D are buffered by the kernel. With the standard ping test, the buffer was no longer able to keep up after 95% of the packets had been transmitted. With the ping flood test, this occurred at approximately 82% of packets transmitted. Native implementation of MT6D in the network stack could address this issue. Expanding the kernel buffer might also address this problem, but would likely result in increased latency.

The next set of tests were designed to evaluate MT6D's ability to pass connection-oriented traffic. We transmitted files ranging in size from 1KB to 500MB and measured packet loss. There was no packet loss for files 100KB and below. This is because the the files were small enough to not fill the kernel buffers. Files greater than 1MB experienced the same group packet loss that we observed with connectionless traffic. These lost packets were successfully retransmitted by the endpoints using TCP, regardless of the number of address rotations.

B. *Latency*

To measure latency, we calculated the amount of time it took for packets received by an MT6D gateway at points 1 or 4 to be retransmitted at points 2 or 3, respectively. The latency incurred during periods where no address rotation occurred was approximately three milliseconds. During an address change the latency was greater, approximately 12 milliseconds, due to the temporary disabling of the NIC during address binding.

The latency added to MT6D-enabled communications can be traced to three sources. First, our prototype implementation required that all packets be decoded and routed in the application's memory. These operations put a significant load on both the system and the MT6D application. Second, the prototype was developed using Python. Python is an interpreted language and is ill-suited to perform the kind of real-time network switching that is required by MT6D. We acknowledge that ultimately MT6D would be best implemented in a lower-level language or hardware. Finally, rotating addresses cause the OS to temporarily disable the NIC and drop packets. Despite the latency experienced during our tests, large TCP sessions were able to recover seamlessly during address rotations without any need for the MT6D device to reconfigure or restart a session. As a case in point, the 500MB file transfer rotated through over 500 address pairs without losing the session.

VIII. LIMITATIONS

MT6D is an extremely capable security mechanism, however, it does have limitations. One limitation is that MT6D is designed to operate on an IPv6 network. The concept of address rotation would work in IPv4; however, the majority of IPv4 subnets do not contain enough free space to facilitate it. Even if a pool of IPv4 addresses were reserved for address rotation, it would be trivial for an attacker to locate a host through simple scanning. An exhaustive scan of a /16 subnet in IPv4 could be accomplished by a single host in three hours or less [6]. We do suspect that MT6D will function through a 6to4 tunnel [18] with a decrease in the maximum transmission unit (MTU) to accommodate the additional IPv4 header. This capability will be tested in a future version of MT6D.

The latency and data overhead created by MT6D are also limitations. Due to the encapsulation in the MT6D tunnel, an additional 62 bytes are added to each communication. Since Smart Grid communications are more likely to contain small pieces of information and will not exceed the link MTU, this additional data overhead will not have any impact on throughput. Also, since the packets must be tunneled and encrypted, there is latency added to process the packets. Yet, all security devices add some latency to systems. As long as the latency added by MT6D lies within the defined tolerances to maintain real time communications, this latency should be acceptable.

MT6D was designed to operate in an IPv6 network using stateless addresses. MT6D could not operate in a network implementing stateful addressing through DHCPv6 without modification to its design. It is unlikely, however, that many networks will implement DHCPv6 due to the management overhead. If this trend changes in the future, we will explore the modifications needed for MT6D to operate in a stateful environment.

Other than normal packet loss, there are a few scenarios where MT6D could drop packets. The first, and most likely, scenario is when a packet gets delayed past the address rotation time. A packet arriving past an address rotation will be dropped by design. This issue occurs in our prototype primarily due to buffering in software. This should be less of an issue in a hardware version of MT6D. A second issue can occur if there is an address collision with an advertised address. The other communicating host has no way of knowing about this collision. Therefore, all packets sent during the rotation interval will be dropped. The likelihood of an address collision, however, is very small. The probability can be written as: $P_c = h/2^{64}$ where P_c represents the probability of a collision and h represents the number of other hosts on the subnet. To minimize the number of packets lost in the unlikely event of an address collision, the address rotation interval can be made shorter. Reducing the interval between address rotations will also increase the computational requirement.

IX. Future Work

Instead of relying on a custom or independent public key infrastructure (PKI) solution to secure the Smart Grid, analysis must be done to determine how to integrate Smart Grid security solutions with the emerging Federal ID Plan. The federal government is looking for private sector to drive identity-based security solutions. The emerging market of the Smart Grid has the potential to drive these new security solutions, since most citizens pay personal electric utility bills. Leveraging the Smart Grid to jump start the Federal ID Plan, security through an identity-based PKI solution, both enhances security for the Smart Grid and contributes to the future of PKI solutions.

To analyze the effectiveness of using MT6D and IPv6 in a Smart Grid deployment, a case study of Landis+Gyr deployment in Texas should be completed. Landis+Gyr already has a pilot deployment of a Smart Grid and uses IP-based communications systems. The meters default to IP-based communications over IPv6. As IPv6 begins to penetrate the consumer Internet Service Provider (ISP) market, using these meters in their default communication mode would be possible without building additional infrastructure. Adding separate MT6D hardware to the current deployment could allow for the limitations and performance of IPv6 to be analyzed for effective use in an actual Smart Grid deployment.

X. Conclusion

Security in the Smart Grid is not only important to securing the new communications and systems on the Internet, but also to ensuring safety and reliability for the critical utility of power. The combination of the immense address space of IPv6 and the security of a moving target defense make MT6D a viable solution for security in the Smart Grid.

References

[1] U.S. Department of Energy, "The smart grid: an introduction," 2008. [Online]. Available: http://www.smartgrid.gov/sites/default/files/pdfs/sg_introduction.pdf

[2] I. van Beijnum, "River of IPv4 addresses officially runs dry," *Ars Technica*, Feb. 2011.

[3] "U.S. & World population clocks," Available at: http://www.census.gov/population/www/popclockus.html/ accessed on 4 Mar 2010.

[4] A. Metke and R. Ekl, "Security technology for smart grid networks," *Smart Grid, IEEE Transactions on*, vol. 1, no. 1, pp. 99 –107, Jun. 2010.

[5] A. Cavoukian, J. Polonetsky, and C. Wolf, "SmartPrivacy for the smart grid: embedding privacy into the design of electricity conservation," *Identity in the Information Society*, vol. 3, no. 2, pp. 275–294, 2010.

[6] S. Groat, M. Dunlop, R. Marchany, and J. Tront, "Using dynamic addressing for a moving target defense," in *the 6th International Conference on Information Warfare and Security (ICIW 2011)*, Mar. 2011.

[7] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)," RFC 4861 (Draft Standard), Internet Engineering Task Force, Sep. 2007, updated by RFC 5942. [Online]. Available: http://www.ietf.org/rfc/rfc4861.txt

[8] D. N. Serpanos and R. Giladi, *Security and Embedded Systems: Volume 2 NATO Security through Science Series: Information and Communication Security (Nato Security Through Science)*. IOS Press, Inc., 2006.

[9] T. Narten, R. Draves, and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," RFC 4941 (Draft Standard), Internet Engineering Task Force, Sep. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4941.txt

[10] M. Bagnulo and J. Arkko, "Cryptographically Generated Addresses (CGA) Extension Field Format," RFC 4581 (Proposed Standard), Internet Engineering Task Force, Oct. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4581.txt

[11] M. Dunlop, S. Groat, W. Urbanski, R. Marchany, and J. Tront, "MT6D: a moving target IPv6 defense," in *the 2011 Military Communications Conference (MILCOM), to appear*, Baltimore, Maryland, Nov. 2011.

[12] S. Groat, M. Dunlop, R. Marchany, and J. Tront, "Privacy and security of DHCP unique identifiers," *International Journal for Information Security Research (IJISR), in press*, vol. 1, no. 4, 2011.

[13] P. Koopman, "Embedded system security," *Computer*, vol. 37, no. 7, pp. 95 – 97, Jul. 2004.

[14] T. Martin, M. Hsiao, D. Ha, and J. Krishnaswami, "Denial-of-service attacks on battery-powered mobile computers," *Pervasive Computing and Communications, IEEE International Conference on*, vol. 0, p. 309, 2004.

[15] G. Jacoby and N. Davis, "Battery-based intrusion detection," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, vol. 4, Nov. 2004, pp. 2250–2255.

[16] D. Ghosh, R. Sharman, H. R. Rao, and S. Upadhyaya, "Self-healing systems – survey and synthesis," *Decision Support Systems*, vol. 42, no. 4, pp. 2164 – 2185, 2007.

[17] M. Amin, "Challenges in reliability, security, efficiency, and resilience of energy infrastructure: Toward smart self-healing electric power grid," in *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*, Jul. 2008, pp. 1 –5.

[18] B. Carpenter and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds," RFC 3056 (Proposed Standard), Internet Engineering Task Force, Feb. 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3056.txt