Moving target defense for securing smart grid communications: Architectural design, implementation and evaluation

by

Aswin Chidambaram Pappa

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Engineering

Program of Study Committee:
Manimaran Govindarasu, Major Professor
Ahmed E. Kamal
Venkataramana Ajjarapu

Iowa State University

Ames, Iowa

2016

DEDICATION

Dedicated to my beloved Sri Radha Madhana Gopal

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

NOMENCLATURE

ACE	Area Control Error
AGC	Automatic Governor Control
APT	Advanced Persistent Threat
ASLR	Address Space Layout Randomization
AV	Anti-Virus
CC	Control Centre
CI	Critical Infrastructure
CIP	Critical Infrastructure Protection
CPS	Cyber Physical Systems
CPU	Central Processing Unit
DNAT	Destination Address Translation
DNP	Distributed Network Protocol
DoS	Denial of Service
FDM	Frequency Division Multiplexing
GRE	Generic Routing Encapsulation
ICS	Industrial Control Systems
IDS	Intrusion Detection System
IEDs	Intelligent Electronic Devices
IP	Internet Protocol
IPsec	Internet Protocol Security tunneling protocol
ISP	Internet Service Provider

ISR	Instruction Set Randomization
MAC	Media Access Control
MitM	Man in the Middle
MS	Master Station
MTD	Moving Target Defense
NAT	Network Address Translators
NERC	North American Electric Reliability Corporation
NIC	Network Interface Card
OLE	Object Linking and Embedding
OPC	OLE for Process Control
OS	Operating System
OSI	Open Systems Interconnection
P2P	Point-to-Point communication
PAT	Port Address Translation
PMUs	Phasor Measurement Units
QoS	Quality of Service
RAS	Remedial Action Scheme
RTT	Round Trip Time
RTUs	Remote Terminal Units
SCADA	Supervisory Control And Data Acquisition
SDH	Synchronous Digital Hierarchy
SE	State Estimation
SNAT	Source Address Translation

SONET	Synchronous Optical Networking
SSLsec	Secure Socket Layer security tunneling protocols
UDP	User Datagram protocol
VM	Virtual Machine
VPN	Virtual Private network
VSAT	Very Small Aperture Terminal
WAN	Wide Area Network

ACKNOWLEDGMENTS

Firstly, I would like to thank the Almighty Divine Lord for His constant mercy and providing me with all the strength and intelligence in pursuing my Master’s degree and teaching me other important lessons for my life. I would like to thank my parents, sister and grandfather whose constant love and support helped me during adverse conditions. I also would like to thank my mentor, Ankur Prabhuji, for his transcendental advice, patience and devotion in helping me solve most of my problems, and providing moral support.

I would like to very much thank my major advisor, Dr. Manimaran Govindarasu, for giving me the great opportunity and honor to be a part of his team. I would like to sincerely appreciate his priceless service of providing me the best academic & research guidance, and constantly motivating me to do a quality and innovative research for the betterment of the society and future students. I owe a great depth of gratitude for his constant encouragement and belief in my abilities to help me achieve things which I had never dreamt of.

I would like to give a hearty salute to my big brother Aditya Ashok, without whom this project would not have taken shape. His selfless dedication to research and leadership qualities are a great zeal of motivation for everyone in our team. And, I would definitely like to thank my other research mates Sujatha, Matt Brown, Bruce, Ryan Goodfellow and Vivek Singh for their constant support for my progress.

I would like to thank DHS and NFS for funding this work under the project “CPS: Synergy: High-Fidelity, Scalable, Open-Access Cyber Security Testbed for Accelerating Smart Grid Innovations and Deployments” (NSF Award # CNS 1446831)

I would like to thank my childhood friends Varshini, BalaKrishnan, GokulaKrishnan, Madhana Gopal, KalaiSelvan and other friends Rajesh, Abhinav, Kushal, Ravi Prabuji and many others for their affection and constantly cheering me.

I would also like to thank Dr. Ahmed Kamal, Dr. Venkataramana Ajjarapu and Dr. Doug Jacobson for consenting to be part of my Program of Study committee.

Finally, I would like to thank my other colleagues, the department faculty and staff for making my time at Iowa State University a wonderful experience. I want to also offer my appreciation to those who were willing to participate in my surveys and observations, without whom, this thesis would not have been possible.

ABSTRACT

Supervisory Control And Data Acquisition (SCADA) communications are often subjected to various kinds of sophisticated cyber-attacks which can have a serious impact on the Critical Infrastructure such as the power grid. Most of the time, the success of the attack is based on the static characteristics of the system, thereby enabling an easier profiling of the target system(s) by the adversary and consequently exploiting their limited resources. In this thesis, a novel approach to mitigate such static vulnerabilities is proposed by implementing a Moving Target Defense (MTD) strategy in a power grid SCADA environment, which leverages the existing communication network with an end-to-end IP Hopping technique among the trusted peer devices. This offers a proactive L3 layer network defense, minimizing IP-specific threats and thwarting worm propagation, APTs, etc., which utilize the cyber kill chain for attacking the system through the SCADA network. The main contribution of this thesis is to show how MTD concepts provide proactive defense against targeted cyber-attacks, and a dynamic attack surface to adversaries without compromising the availability of a SCADA system.

Specifically, the thesis presents a brief overview of the different type of MTD designs, the proposed MTD architecture and its implementation with IP hopping technique over a Control Center–Substation network link along with a 3-way handshake protocol for synchronization on the Iowa State’s Power Cyber testbed. The thesis further investigates the delay and throughput characteristics of the entire system with and without the MTD to choose the best hopping rate for the given link. It also includes additional contributions for making the testbed scenarios more realistic to real world scenarios with multi-hop, multi-path

WAN. Using that and studying a specific attack model, the thesis analyses the best ranges of IP address for different hopping rate and different number of interfaces. Finally, the thesis describes two case studies to explore and identify potential weaknesses of the proposed mechanism, and also experimentally validate the proposed mitigation alterations to resolve the discovered vulnerabilities. As part of future work, we plan to extend this work by optimizing the MTD algorithm to be more resilient by incorporating other techniques like network port mutation to further increase the attack complexity and cost.

CHAPTER I

INTRODUCTION

Smart Grid and other Cyber Physical Systems (CPS) use Supervisory Control and Data Acquisition (SCADA) network as an essential backbone for monitoring, controlling and protecting the Critical Infrastructure (CI) resources incorporated within the system. The architecture and the protocols in a SCADA system is shown in figure 1. Several cyber-attacks in the recent past indicate an increasing trend of sophisticated adversaries exploiting the static system configuration and the underlying vulnerabilities present in these legacy computational and networking system resources [17], [19]. Also the Smart Grid consists of several types of embedded devices with potentially vulnerable firmware, which if left unpatched might be an easy target for attackers to create backdoors and perform sophisticated automated attacks like Stuxnet worm [17], which is a rootkit exploiting the Siemens Programmable Logic Controllers to destroy the centrifuges inside Iran's Natanz uranium enrichment facility.

However, even if these devices were secured by traditional security devices like Firewall, IDS/IPS, antivirus, etc., still they are not completely free from attacks which can bypass these security measures. The main reason being the presence of static configurations which makes the devices a sitting duck for the attackers to shoot down. Also, even some of these security devices lack the intelligence to detect and protect against more complicated attacks like coordinated attacks.

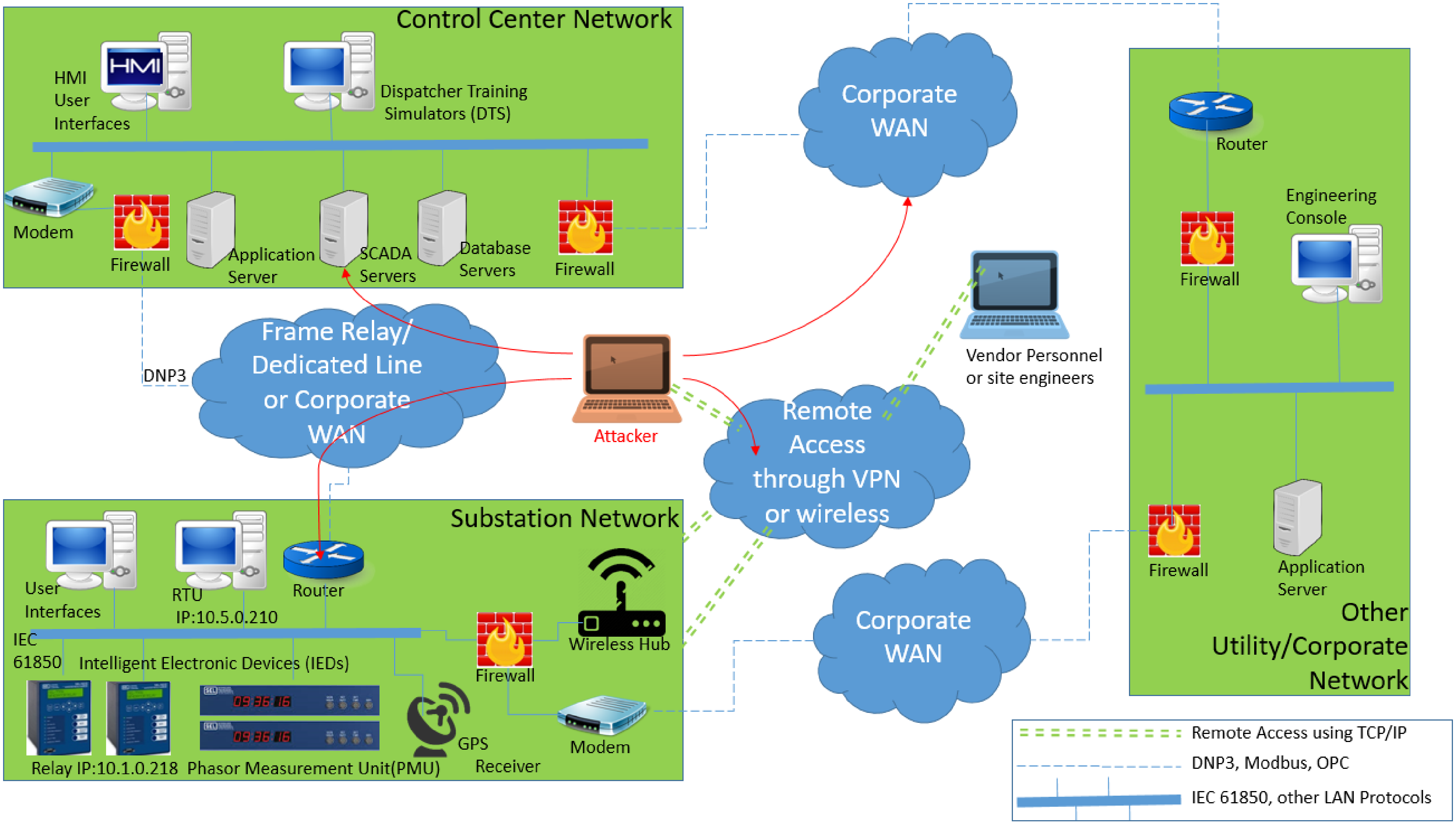


Figure 1. *SCADA Architecture and Protocols*

For example, an attacker targeting and flooding a particular device in the SCADA network can be prevented by whitelisting or IP filtering the unauthorized IPs, but if the attacker spoofs the authorized IPs for flooding the targeted system which is incapable of handling the overload, it may crash and lead to a Denial of Service (DoS). Even if we think of throttling the traffic to avoid system crashes, it might result in dropping of legitimate packets, still leading to a DoS, which could have a huge impact in case of CPS such as the power grid. Alternatively, a system with dynamic network and computing resources or configurations can prevent some of these attacks, by dynamically varying the topology the system to confuse the attacker. At the same time, such a defense system should also ensure proper synchronization and availability to authorized communication nodes.

This thesis introduces a novel Moving Target Defense (MTD) technique for a SCADA network, which uses an IP-Hopping strategy to mutate the IP addresses of the gateway router's external interface IPs providing transparent and reliable services to the end systems and without needing to perform any additional configuration on them, thereby avoiding operational overheads at system level. Thus, it prevents attackers from targeting the internal network as the IP addresses used by the gateway MTD routers are hopping dynamically and randomly, preventing the attacker to target the victim and discover their vulnerabilities. We implement this defense strategy in a traditional SCADA system consisting of Control Center and Substation network with physical resources such as protective relays. The IP-Hopping algorithm makes use of a random IP generator function, which generates random IP address in a defined range of subnets or multiple public address ranges owned from the Internet Service Provider (ISP) providing routing path mutation. This method not only prevents attackers from targeting the system’s cyber resources, but also actively filters out those unwanted traffic which was not destined to the current random IP address. It also provides uninterrupted service by synchronization using a 3-way handshake protocol.

The proposed MTD technique has been implemented and evaluated in Iowa State University’s Power Cyber testbed [18]. As part of the experimental validation, the throughput and delay characteristics were analyzed for different IP hopping rates to tune the MTD mechanism for the best possible performance for the given SCADA communication links. We also analyzed the limitations of the proposed MTD algorithm and provided additional mitigation methods to overcome those limitations.

One of the main contributions of this thesis is to highlight the feasibility of implementing a MTD strategy in a realistic SCADA environment such as the Power Cyber testbed. The proposed algorithm not only provides a dynamic attack surface to prevent certain types of targeted attacks, but also ensures seamless connectivity to support traditional SCADA operation for power grid monitoring and control.

1.1 Smart Grid Communications

1.1.1 SCADA communication links:

Industrial Control Systems (ICS), such as the Power Grids use SCADA Communications to monitor and control physical systems like power generation, water distribution, oil and gas pipelines, traffic lights etc. The conventional communication links adopted in SCADA systems exerts wireless and wireline technologies, along with other hybrid facilities like Synchronous Optical Networking/Synchronous Digital Hierarchy (SONET/SDH), which were mostly used for railways and power stations. SCADA traffic is usually casted over corporate network established either by ISPs or with dedicated links established by the industry.

1.1.2 SCADA protocols:

SCADA protocols are application specific, and differ a lot from the normal IT traffic. They are precise and carry only the information which is polled by the Master Station (MS) in the Control Centre (CC). Usually the communication happens between the CC and the Substation Network, or between the Remote Terminal Units (RTUs) in the Substations. The RTUs in the Substations aggregate and locally process the measurement data from the physical devices like Relays, Phasor Measurement Units (PMUs), etc. The RTUs, PLCs and other remote controller systems were initially designed to manage the variety of communication protocols used by the various Intelligent Electronic Devices (IEDs). Modbus, Profibus and RP-570 are the conventional and commonly used SCADA Protocols which are more vendor specific. Whereas the currently used ones are DNP3, IEC 61850 and IEC 60870-5-101 or 104. They are regulated and sanctioned by most SCADA vendors and also they can be encapsulated over TCP/IP. Modern SCADA vendors also use OLE for Process Control (OPC) for intercommunicating different software and hardware, making scalable interoperability between various industrial products to be a part of the SCADA control loop.

1.1.3 Smart Grid cyber security requirements:

The security requirements for a Smart Grid control communication is standardized by the North American Electric Reliability Corporation (NERC) and Critical Infrastructure Protection (CIP) in the US. As there is more expectation for security requirements from them in the recent days due to increased cyber-attacks, there is also an increasing use of satellite-based communication. The main benefit of this satellite communication is that it is more reliable and can incorporate its own crypto mechanism, without relying on the ISP networks. Though previous services provided by Very Small Aperture Terminal (VSAT) were of low quality, newer carrier-class systems supply enough bandwidth and minimum latency for the effective functioning of SCADA operations, without any service unavailability issues.

1.2 Thesis Motivation

1.2.1 Static Cyber Threats:

The system and network configurations in current Smart Grid infrastructure are static from an attacker's point of view. Even if the configurations are updated, they are in a slow pace giving the attacker more time to trace out the updated configurations for creating a successful attack. The conventional security schemes just focus in protecting these static resources from the attackers' target. But, when the system gets very huge and especially when Critical Infrastructures are involved, the threat involved is huge. Even if there is a small loop hole, it is sufficient for the attacker to seep in to establish a trapdoor, and gradually exploit the vulnerabilities present in the inside system to execute a impactful attack. Security experts are always looking for directions to minimize the cyber risk, which is calculated by the following equation,

$$\text{Cyber Risk} = \text{Threats} * \text{Vulnerabilities} * \text{Consequences}$$

But unfortunately, the threat modelling is not always accurate and hence there may always exist uncertain and unknown threats, which might be missed out. Also, we have approximately 65K vulnerabilities in the current CVE Database. It is too much time consuming and computationally inefficient to check for the system behavior for all these vulnerabilities, in such a huge and complex systems. Even with all these checks and rules, still things are unsure because of false positives. Therefore, achieving a 100% secure system is always theoretical. But it is smarter to confuse an attacker and prevent an attack using a proactive defense technique in addition to having a reactive type of defense scheme.

Observing the statistics of cyber-attacks, intrusions are most cases inevitable and most breaches are discovered by third parties. In many cases, malwares are already installed inside the system and the intruders stay in the systems for days, weeks and even for months.

In order to come up with a better defense scheme, it is always recommended to first have a good understanding of the attacker's methodology to think like a hacker. The adversaries, whether he be a hacktivist or even a nation-state hacker, they usually proceed step by step through the Cyber Kill Chain in order to execute a complex and successful attack. The figure 2 shows the different steps involved in a cyber-attack, from step-1 (reconnaissance) through step-5 (persistence).

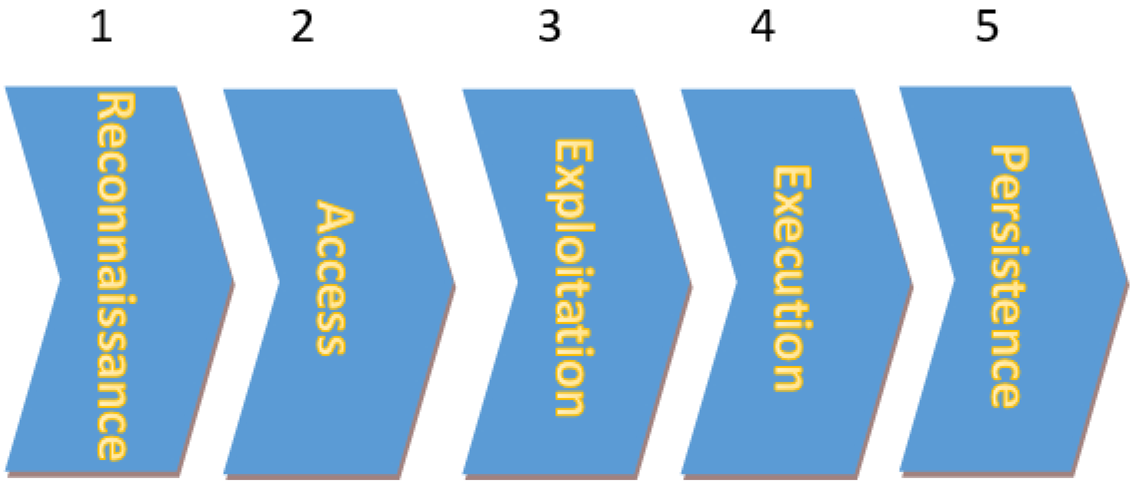


Figure 2. *Cyber Kill Chain*

1.2.2 Cyber Kill Chain

1. Reconnaissance:

   Initially the attacker collects useful information about the target by using scanning tools like Nmap, Zenmap, etc. He develops a blueprint of the system architecture with Hostname-IP address maps and an up-to-date network diagram and other activity logs.

2. Access:

   After reconnaissance, the attacker tries to connect or communicate with the target to identify its properties like version numbers, vulnerabilities, and configurations, etc., using host deep/ port scanning tools and discovering vulnerabilities using vulnerability assessment tools like Nessus, Owasp, etc.

3. Exploit Development:

   This is the weaponization phase. After the attacker finds a vulnerability in the system, he develops an exploit for that vulnerability in order to gain a foothold or escalate his privilege to launch a successful attack. He might use tools like metasploit for this purpose to develop attack scripts or malware for specific types of attacks.

4. Attack Launch/ Execution:

   Once, the attacker has sufficient and necessary privilege to execute commands in the system, he delivers the exploit to the target, and executes the attack script or malware. This can be either through a network connection or using phishing or using a well sophisticated supply chain or gap jumping at attack like through an infected USB pen-drive.

5. Persistence:

   Usually this is an optional phase where the attacker installs additional backdoors or access channels to keep his persistence access to the compromised target system. Or this might help him launch further more impactful attack in the future through repeating above steps. The adversaries also enable lateral movement and exfiltration of data to facilitate memory leakages.

Making any one of these phases difficult, can help to thwart the cyber-attack. It is better if the first phase, Reconnaissance is made difficult for an attacker, i.e., to make it difficult for an attacker to scan and collect information on the topology of the SCADA network. This is where Network-based MTDs can have a role in dynamically changing the network parameters and topology making it impossible for an attacker to guess or know the exact blueprint of the topology.

1.2.3 Problem Identification using a specific attack scenario:

Consider a scenario as shown in the figure 3. below, which shows the most common architecture of a Control Center – Substation SCADA network. The substation network (on the top right) and the Control Center network (on the bottom right) are connected over the Wide Area Network (WAN). Since the WAN is exposed to outside world, it is more likely that the traffic is subjected to attacks. Considering a scenario, where the attacker eavesdrop the traffic and finds out that DNP3 communication is established between the Substation RTU and Control Center SCADA Server. Analyzing the eavesdropped packets, he extracts the IP addresses of both the machines, and targets one of them (say RTU 10.5.0.210) for attacking the system (say tripping a breaker system 10.1.0.218, as the Relays are connected to and controlled by the RTU). Now the attacker can just replay the legitimate trip command issued by the Control Center Master Station at a different point of time by selecting a different breaker in the system to maliciously trip open a Relay.

From a conventional security best practice, this type of attack can be easily eliminated by adding a firewall or IDS rule on the Substation gateway router or substation machine to detect and drop Trip packets coming unauthenticated source IP addresses.

But, what if the attacker is smart enough to masquerade the source IP address of the legitimate system. In this instance, just a firewall rule will fail to drop the Trip command packet as it has proper authentication. A distributed IDS on the both the Substation and Control Center network with event correlations can detect such an attack, but cannot prevent it as it is too costly and time consuming to check for each and every events, delaying the normal legitimate communication beyond the acceptable limit for a SCADA network, and making the system prone to unavailability or DoS issues.

Since, the Availability of the system is of highest priority and concern in a SCADA infrastructure, a distributed IDS fails to prevent such an attack at an early stage. May be it can detect the attack at a little later point of time and help the system to recover after the attack, but is not a very wise design.

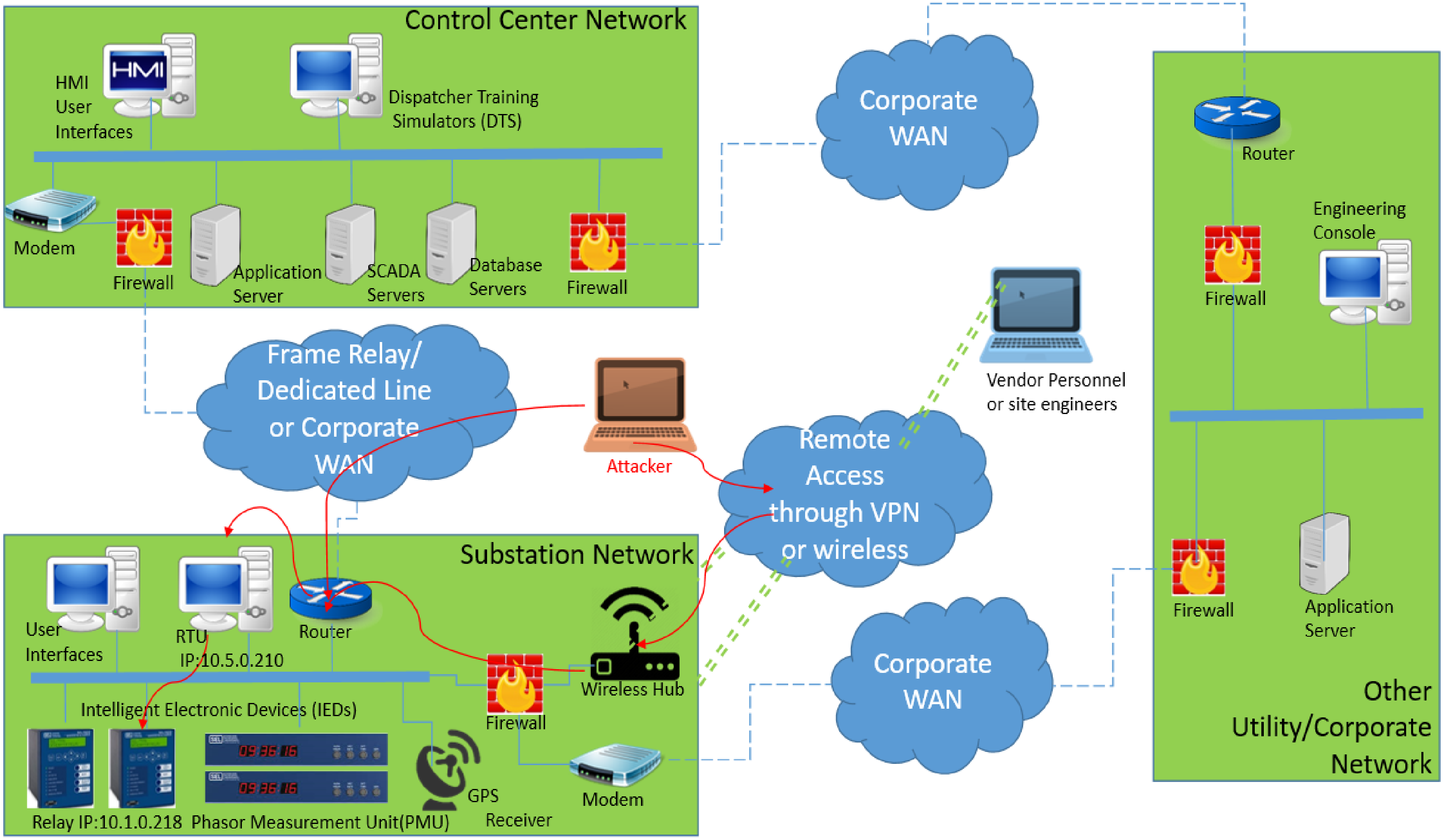


Figure 3. Scenario describing the security problems with static configurations

1.2.4 Need for a Dynamic Defense Strategy – Network based MTD:

Now, a Network based Moving Target Defense with dynamically changing IP addresses of the SCADA machines installed on both the networks, by the Gateway routers, will prevent an attacker from targeting the machines, thereby making him unable to continue with the next consecutive steps of vulnerability discovery, exploitation and attack execution. Since the SCADA machines usually don't need to be connected to the internet, this design is more feasible, and other than the IP switching delays, the SCADA traffic suffers no other processing delays at the end point gateways. Therefore, having a network based MTD could possibly prevent a large number of cyber-attacks which are propagated and permeated through the network.

This thesis will give a detailed description on the different design types of MTDs, the proposed IP hopping MTD for a SCADA infrastructure, and the experimental analysis of the effectiveness of the proposed scheme in terms of throughput and delay latency, and some experimental evaluations to select the parameters for IP hopping like the best hopping rate and the best set & range of IP addresses required for the reliable and resilient functioning of the MTD. Later, it discusses some scenarios that limits the proposed design and it gives appropriate solutions for optimizing the proposed algorithm. Finally, the thesis presents new directions for future research with conclusions.

1.3 Related Works

Department of Homeland Security (DHS), has funded projects in Cyber Security Division for researching and developing various MTD themes through its prime performers namely, Florida Institute of Technology (FIT) – Federated Command and Control (FC2), Carnegie

Mellon University/ Software Engineering Institute (CMU / SEI) – Moving Target Reference Implementation, Def-Logix – Hardware Enabled Zero Day Protection (HEZDP), IBM – Hardware Support for Malware Defense and End-to-End Trust and Princeton University – Newcache.

The Cyber Security lab, Argus, in University of South Florida, are also working on the MTD research [29], [30], [31], [32], [33], [34], [35] & [36], and have released a software product Ancor, which is a cloud automation framework that encompasses dependencies between various layers in a cloud based IT system to be mapped into an API stack. The changes made in the instances of one layer are reflected using the software to reconfigure instances in other layers, creating a MTD platform for the systems being deployed and managed by it.

Atighetchi et al. [1] presented a network-centric port and address hopping mechanism based on random number generation and time synchronization using Network Address Translation (NAT) to map True IP-port and False IP-port dynamically.

Jafarian et al. [3] developed an OpenFlow Random Host Mutation (OF-RHM) IP address randomization scheme for Software Defined Networking (SDN) based on Open Flow controller and virtual IPs.

Wang et al. [4] propose MOTAG, a MTD mechanism Against Internet Distributed Denial-of-Service (DDoS) attacks using SDNs. Chavez et al. [9] introduced three techniques for the dynamic randomization of network attributes of Critical Infrastructure systems by dynamically recon-figuring the network settings. They also used SDNs to randomize TCP/UDP ports, IP addresses and Network paths. Nevertheless, the IP randomization was implemented only at the switch level and restricting its scope to Wide Area Networks

(WAN). Also, all the above techniques cannot be used in a legacy networking infrastructure which does not support SDN and could be used only where SDNs are supported such as in cloud computing and enterprise IT networks.

Al-Shaer et al. [5], Jafarian et al. [6] and Luo et al. [8] presented a dynamic address hopping mechanism. Luo et al. in addition used dynamic port hopping mechanism called Random Port and Address Hopping (RPAH), which mutates IP and port addresses with respect to time, source and service identity to thwart against both internal and external adversaries like scanning, SYN flooding and worm propagation attacks. However, the operational overhead and delay latency introduced by these techniques are unacceptable to specific SCADA applications under the umbrella of power grid monitoring and control.
Li et al. [10] presented a traffic morphing algorithm, CPSMorph, to protect CPS network sessions against traffic attacks, by morphing the distributions of inter-packet delays to make the CPS sessions statistically indistinguishable from those of typical network sessions, considering the time constraints into account.

Although, Groat et al. [16] proposed a proactive net-work layer Moving Target IPv6 Defense (MT6D) technique for securing Smart Grid communications using the vast subnet address space offered by the new Internet Protocol version 6 (IPv6), the MT6D testbed represents a client-server model more similar to an IT environment than compared to a CPS system, thus making the results less accurate for SCADA systems. Also, the MT6D testbed was evaluated with non SCADA traffic (just ping and HTTP over TCP sessions), thus neglecting the latency sensitive nature of a CPS communication infrastructure.

To the best of our knowledge, none of the above works have specifically analyzed the suitability and feasibility of implementing MTD strategies on a realistic SCADA environment. In the reminder of this thesis, we describe how the proposed MTD IP Hopping scheme can help defend a realistic power grid SCADA system against certain types of data integrity [12], command and control and network traffic attacks [18].

1.4 Thesis Organization

The remainder of this thesis is organized as follows:

- **Chapter 2** compares and analyzes the various cyber defense strategies for a CPS used in a SCADA infrastructure. An intuitive analysis presents the attack/defense comparison as well as the performance comparison of all the major cyber defense solutions.
- **Chapter 3** reviews and presents the different types of MTD designs and explains the architecture of the MTD experimental setup on the Power Cyber testbed and describes the design and implementation of the IP hopping algorithm and the 3-way handshake protocol for synchronization between IP hopping devices.
- **Chapter 4** explains in brief the various implementation methods of deploying a Network based MTD in a SCADA infrastructure and presents experimental evaluation of different SCADA communication metrics for effectiveness and better performance and security of the proposed algorithm with additional insights on realistic scenarios. It also briefly discusses various other special attack scenarios and provides mitigation methods for addressing limitations of the proposed IP hopping strategy.
- **Chapter 5** concludes the thesis with a summary and potential directions for future work.

CHAPTER 2

COMPARATIVE ANALYSIS OF VARIOUS CYBER DEFENSE TECHNIQUES FOR A SCADA INFRASTRUCTURE

The Cyber Physical System's security requirements are quite different from the conventional IT security requirements. Table 1. shows the priority levels of CIA/AIC triad model between an IT and a SCADA infrastructure.

Table 1. *CIA/IAC triad priority levels of IT/SCADA infrastructure*

Priority	**IT**	**SCADA/ICS**
#1	Confidentiality	Availability
#2	Integrity	Integrity
#3	Availability	Confidentiality

From the above table, we see that the availability and integrity of the system is of major concern in a Smart Grid system than compared to the IT infrastructure, which gives more importance to the Confidentiality of information.

There are various Cyber defence strategies installed in a Smart Grid communication to prevent different types of outsider as well as insider attacks. Among them here we will discuss the most significantly and commonly used ones. Based on their operational behaviour, they can be broadly categorized into four different classes as follows:

1. Secure Protocols : DNP3sec, Secure Modbus, etc.
2. Crypto Encapsulation : VPN/ GRE Tunnelling/ IPsec/SSLsec, etc.
3. End point filters : IDS/ IPS/ Firewall/ AV, etc
4. Obfuscation : MTD

The merits and demerits of these four defence mechanisms should be compared with respect to its defence capabilities in mitigating various kinds of cyber-attacks and their system performance, and should be analysed in context to a Cyber Physical System Security to understand the significance and need for a MTD mechanism.

Some of the pros and cons of these four defence strategies are analysed and listed as follows:

2.1 DNP3sec, Secure Modbus Protocol

These ICS specific secure protocols provide authentication and integrity by verifying the frame origin, and providing confidentiality through encryption techniques.

Pros:

- Low overhead
- Offers Authentication and Integrity
- Application level security compatible with existing standards
- Aggressive Mode: high priority for critical messages (less bandwidth, but less secure)
- Simple and more robust Key Management techniques

Cons:

- Requires some modification to DNP3 Data Link Layer
- Quantum Computing Attacks: cryptanalysis of captured packets
- No encryption, only authentication
- Doesn’t provide Holistic security for SCADA communication (Need to build individual security framework for other security protocols)

2.2 Scalence/ VPNsec/ GRE Tunnelling/ IPSec/ SSL wrapper

These crypto encapsulation/tunneling techniques send the entire communication payload over an encrypted trusted channel pre-established between both the end points (P2P).

<u>Pros:</u>

- Transparent to applications: No application dependence
- Securing Real-time traffic: Encryption prevents eavesdropping
- Inbuilt with IPv6 and emerging technologies
- Available for all common OS, can be installed either in Workstations or service supported by routers/gateways

<u>Cons:</u>

- Broken Algorithms: Frequently needed to be patched, else pose severe threats.
- Not adaptable with deep packet inspection IDS
- No Non-repudiation services
- Latency and Heavy CPU overhead: Performing encryption and decryption at high speed
- Prone to DoS: StaticPort providing VPN service can be easily blocked and hence not reliable
- Prone to Password guessing attacks: Human error & weaknesses
- Requires all intermediate devices to support TCP and UDP communications on port number 20000

2.3 IDS/IPS/firewall

These are the most significant filtering techniques based on manually coded rules or rules synthesized based on system anomalies with certain threshold values to detect and alarm the system when a malicious or unexpected event takes place.

Pros:

- Allows for Incidence Response and Event Management: excellent auditing
- Tracking of virus propagation: Stuxnet
- Response capabilities: Automation of defense after detection
- Versatile in customizing to CPS specific behavior/pattern analysis
- Robust/reliable and efficient in filtering traffic attacks

Cons:

- False Positives and false negatives
- More maintenance and needs active experienced operational support
- Complicated setup and difficult to design a best model
- Cannot monitor/analyze encrypted traffic
- Fails to counter spoofed attacks
- Bypassing firewall/ IDS with new attack vectors (tunneling attacks)

2.4 Moving Target Defense

A random scheme introduced into the normal functioning of a system to obfuscate the understanding of an adversary from learning the system properties.

Pros:

- DoS protection
- Adds redundancy and improves security
- Drastically reduces attack surface
- Suited for Targeted attacks: scanning/probing/backdoors
- Prolongs/deters infected state by creating more time and space for activating/effecting the defense/recovery actions

Cons:

- Single point of failure: Operational Denial of Service
- Cross Platform design problems
- Highly complex design and less resilient to operational disturbances
- Needs High speed processor for faster dynamics and additional system resources increasing the cost

2.5 Attack / Defense Comparison

The figure 4. shows the Attack/Defense comparison based on a 10 point (0-9 graded) scale rated intuitively based on its mechanism to prevent various different types of attacks (like MitM/Hijacking, DoS, IP spoofing/ intrusion, Malicious Command Injection, timing attacks, trapdoors, APTs, Data leakage, targeted attacks, eavesdropping and random attacks).

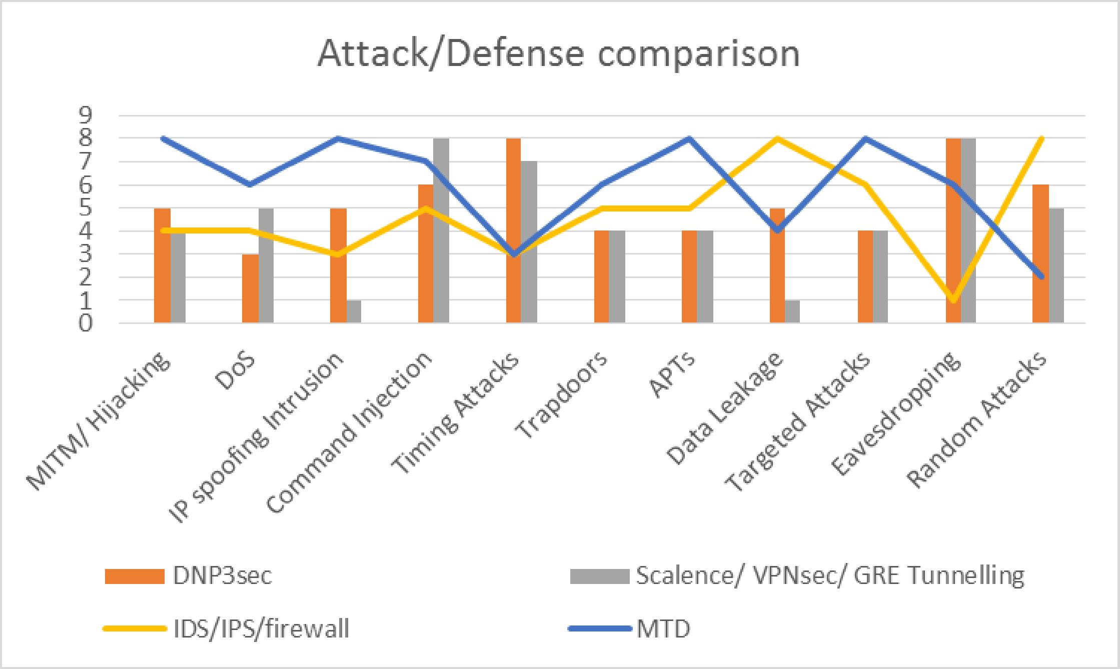


Figure 4. *Attack/Defense comparison*

From the above figure 4, we can observe that the MTD has better performance in various types of targeted attacks like MitM/Hijacking, DoS, IP spoofing, intrusion, command injection, trapdoor, APTs, etc., because it dynamically changes the target of the attacker making the probability of a successful attack very low. But, it has lower defense against timing attacks, data leakage attacks and random attacks, in which places the attacker might be able to guess the critical parameters required for the successful functioning of the MTD. To compensate for the poor defense of MTD to such attacks, it is better to have a combination of other defense strategies in addition to MTD. This way MTD can be a complement to other existing defense strategies rather than be a replacement for them.

The Table 2. shows further varieties of threats addressed and unaddressed by the four different defense strategies.

Table 2. *Security features, threats addressed and unaddressed for various defense strategies in a SCADA environment*

Defense Strategies	**Security Features offered:**	**Threats Addressed:**	**Threats Unaddressed:**
DNP3sec/ Secure Modbus Protocol	Authentication and Integrity check	Spoofing, Modification, Replay, Non-repudiation	DoS, Zero day exploits, Traffic attacks
Scalence/ VPNsec/ GRE Tunneling/ IPSec/ SSL wrapper	Confidentiality, Transparent Application Level Security	IPSec-Application layer attacks	Network layer attacks, Phishing attacks, Stealth DoS, Trapdoors, APTs, Rootkits SSLSec- Application layer attacks
IDS/IPS/firewall	Exfiltration, Monitoring, Anomalous behavior detection, Signature analysis	Malicious traffic filtering,	Masquerade, Authentication attacks
MTD	Reduced attack surface, operational confidentiality, state obfuscation	Targeted attacks, DoS, MitM, Spoofing, Insider threat	Replay and Random attacks

2.6 Performance Comparison:

The Security defence comparison alone doesn't give the overall analysis of the four defence systems, but we need to have a system cost/performance comparison as well to get the full picture. Therefore, these four defence techniques are once again graded statistically and intuitively over a 10-point scale (with 0-low and 9-high) against various performance parameters like system utilization, reliability, bandwidth, Quality of Service (QoS), latency delay introduced, cost, implementation difficulty and additional processing speed and computational resources consumed with the addition of the defence strategies.

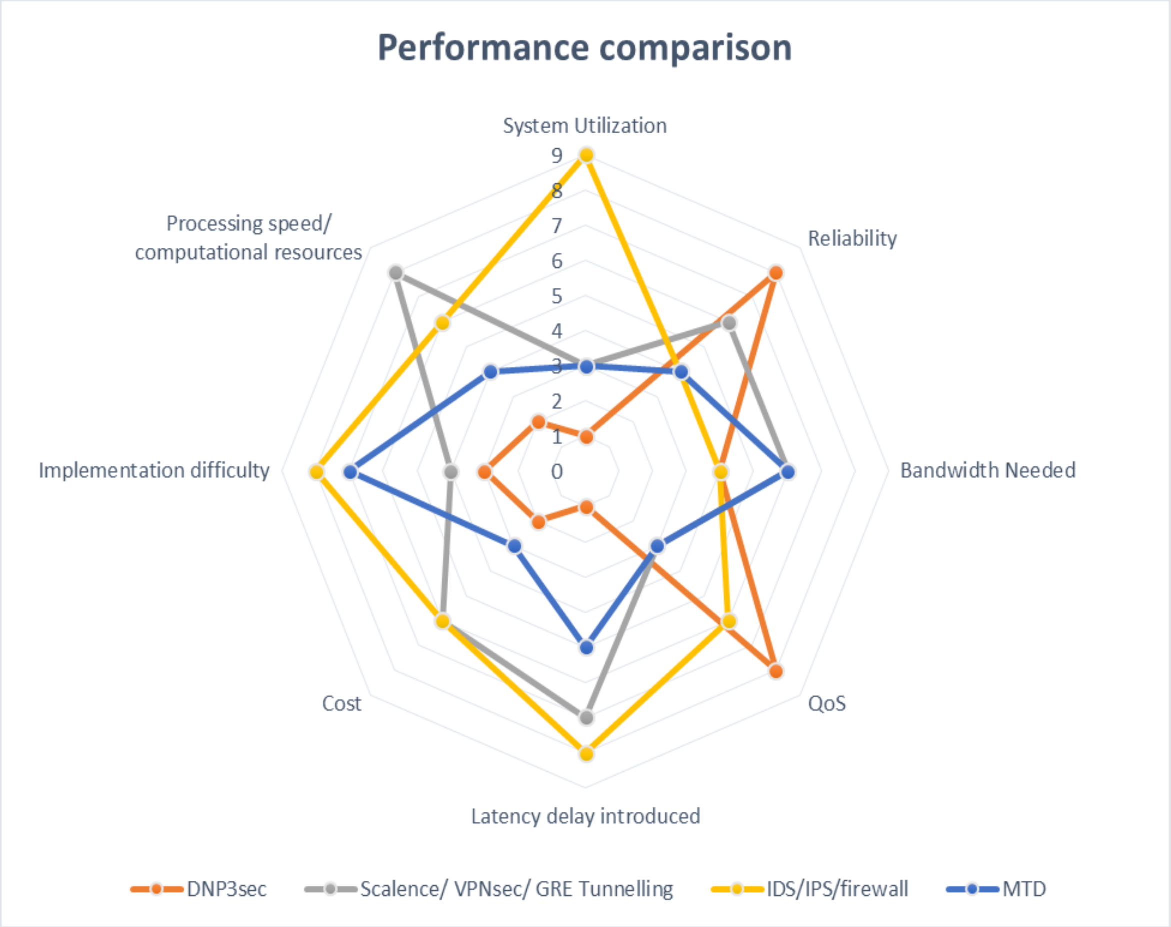


Figure 5. *Performance comparison for various defense schemes*

The above figure 5, shows the radar chart visualization of the performance/cost comparison of the various defense schemes (displayed in different colours), with gradation starting from the lowest (at the centre), extending to the highest (at the circumference). We can infer from the above graph that the MTD (blue line) is more are less equally distributed for all the performance parameters, compared to the other techniques, which are good in certain aspects and low in other factors. Although, the performance is more optimal as per the design of the defense systems, the inherent operational mechanism of the defense system might itself introduce some overheads (like the latency introduced by the VPN or IDS is more as it has to encapsulate or check against all sets of rules in the endpoint gateway systems).The main aim of a MTD is to substantially increase the cost of attacks by deploying and operating networks/ systems/ applications to make them less deterministic, less homogeneous and less static. This can be achieved by continually shifting the system parameters with respect to time to increase the complexity and cost for attackers, by limiting the exposure of attack surface or vulnerabilities and opportunities for cyber-attack, and hence providing proactive system defense. The MTD employs techniques to dynamically alter the system topology and settings in ways that are manageable by the defender yet make the attack space appear unpredictable to the attacker. Thus it can be also referred to in different terms like ‘Cyber Maneuver’ or ‘Adaptive Cyber Defense’, as it involves shifting the defense strategies from reactive to proactive method employing dynamic momentum to the system from its static counterparts.

CHAPTER 3

MTD ARCHITECTURE & ALGORITHM DESIGN

3.1 MTD Design Types

The best MTD design for a particular system needs to provide optimal security to the supporting infrastructure. This is made possible if the defender creates, analyzes, evaluates and deploys suitable mechanisms and strategies to utilize the unused redundant resources of the existing infrastructure to be adapted by the MTD algorithm within safe operational boundaries in ways of providing more defense in depth security. The defense in depth security is usually achieved through a layered approach, where defenders employ different types of security mechanisms at different layers of the OSI model.

Based on the different layers, the design of MTD can be broadly classified into three different categories as follows,

3.1.1 Application-based MTD:

- MTD for State Estimation (SE), where the SE Algorithm at the control center can use different measurements coming from different substations at different instance of time to calculate the SE of the system.
- MTD for Automatic Governor Control (AGC), where the Area Control Error (ACE) values can be X-ored with different values of random values which can be synchronized both at the control center and the substation based on a pre-established random secret seed

- MTD for Remedial Action Scheme (RAS), where the status of the Generator and the generation, transmission or distribution lines can be found out using different combination of measurement sources like PMU readings, Relays, etc.

  Also randomness can be injected in the timings of the packets like inter-packet delay, jitter and increasing the frequency of measurements, provided the network can accommodate for more bandwidth overloads.

3.1.2 System-based MTD:

- Software-based MTDs tries to obfuscate and protect the software against analysis thereby preventing unwanted modification to the source code. Different methods of this involves dynamic runtime environment like Address Space Layout Randomization (ASLR), Instruction Set Randomization (ISR), etc.; dynamic software like in-place code randomization, compiler-based software delivery, etc.; and dynamic data allocation. This prevents the attacker from exploiting the vulnerabilities in the software to execute a stack overflow or buffer overflow attacks, as here the defender randomly chooses the base address of stack, heap, code segments (ASLR), padding stack frames, malloc() calls, location of Global Offset table, etc. Randomization can even be done at compile or link time, or by rewriting existing binaries.
- Hardware-based MTDs, tries to randomize the energy consumptions of certain standard processors, FPGA, etc., by inducing time and algorithmic variations, so that an attacker won’t be able to listen to channels for analyzing and executing attacks like differential power analysis, side channel attacks in Hypervisors, etc. for cracking the cryptographic keys.

3.1.3 Network-based MTD:

- Physical layer MTD involves using various physical channels at random instances like using different types of wireless, wireline, optic lines of communication. It can also use different modulation and demodulation techniques for analog mappings, different frequency bands, and different multiplexing techniques. Within a single multiplexing scheme (say Frequency Division Multiplexing FDM), it can have various altering patterns of frequency hopping for random time slots.
- Datalink layer MTD involves varying Frame lengths, frame structures, with different encoding schemes with various error detection codes. It can also employ varying physical addresses (MAC address) for Media Access Controls.
- Network layer MTD involves changing IP addresses and routing paths, using proxy gateway routers, or Network Address Translators (NAT), or locally changing the packet's source and destination addresses with systematically changing IP-stateful firewall rules mangling the state of connections to reject unwanted traffic (even with legitimate IP address coming at unauthenticated time intervals – intelligent replay attacks)
- Transport layer MTD involves changing session well known Port addresses, with short life ephemeral port numbers and similarly blocking the previous open ports with firewall rules (but this time stateless), to avoid conflicts of existing open sessions and maintaining the seamless continuity of previous session without any service interruptions.

Since, the proposed MTD scheme is based on L3 Network layer, the remaining portion of this chapter will discuss the various implementation styles of L3 Network based MTDs, using conventional networks as well as recent innovative techniques like Software Defined Networks (SDNs).

3.2 Architecture of MTD experimental setup

Figure 6 shows the MTD IP-Hopping architecture for experimental setup. The Control Center consists of two SCADA servers, which run the Siemens Spectrum Power TG Energy Management System (EMS) software, which periodically polls the substation Remote Terminal Units (RTU) for measurements and could also execute commands to control the substation equipment such as protective relays through the DNP3 protocol. The Substation consists of the machines that run the RTU software, which collects analog and status measurement readings from the Physical Relays (orange) connected to on substation internal network.

The entire setup consisting of the Substation network, Wide Area Network (WAN), and the Control Center network have been virtualized and hosted on a machine with VMware workstation, except for the two physical Relays which are connected to the substation’s internal network (VMnet6) by bridging the host machine’s physical NIC externally. The networks inside the workstation are virtually emulated by the VMware workstation and provide realistic network characteristics. The Substation network and the Control Center network are separated by a WAN. Here we assume that the WAN, which is usually provided by an ISP, is susceptible to cyber-attacks. So, here we model an attacker with a virtual machine (VM) (red) running Kali Linux to execute a Man-in-the-Middle (MitM) attack, in order to be able to sniff the traffic and also inject packets between the Substation and the Control Center.

In a normal scenario without the implementation of MTD, the routers (green) for both the Substation and Control Center networks would act as gateways connecting them to the WAN. In such a scenario, the attacker could easily target these gateway routers to penetrate into the either of the Substation or Control Center networks by classical enumeration techniques along the cyber kill chain. They would perform network reconnaissance with scanning tools to look out for vulnerable hosts and services that could be exploited, and create backdoors or propagate worms to inject APTs for executing a successful attack.

Even with traditional end system security techniques like firewall or Intrusion Detection/Prevention Systems (IDS/IPS), the SCADA network is not completely free from attacks as has a static configuration. Moreover, the security mechanisms described above only provide defense against vulnerabilities which have already been discovered leaving them prone to zero-day exploits. In addition, these devices themselves are vulnerable to a Denial of Service attack (DoS) when they are targeted with huge bandwidths of network traffic sent from spoofed source IP addresses that map to legitimate SCADA devices on the network. Therefore, in such scenarios, the traditional static security devices fail to protect the system.

But with the MTD algorithm activated, the gateway routers act as dynamic proxies mutating the IPs of the external interfaces while still providing transparent, uninterrupted and seamless end-to-end SCADA communication. At the same time, the MTD algorithm creates a dynamic network topology making it difficult for the attacker to target the gateway device. Since the attacker doesn’t know the exact IP address of the gateway device to be targeted at a given time instance it becomes harder to perform network reconnaissance and also send malicious traffic to execute successive attacks.

IP Hopping MTD SCADA Tesbed Architecture

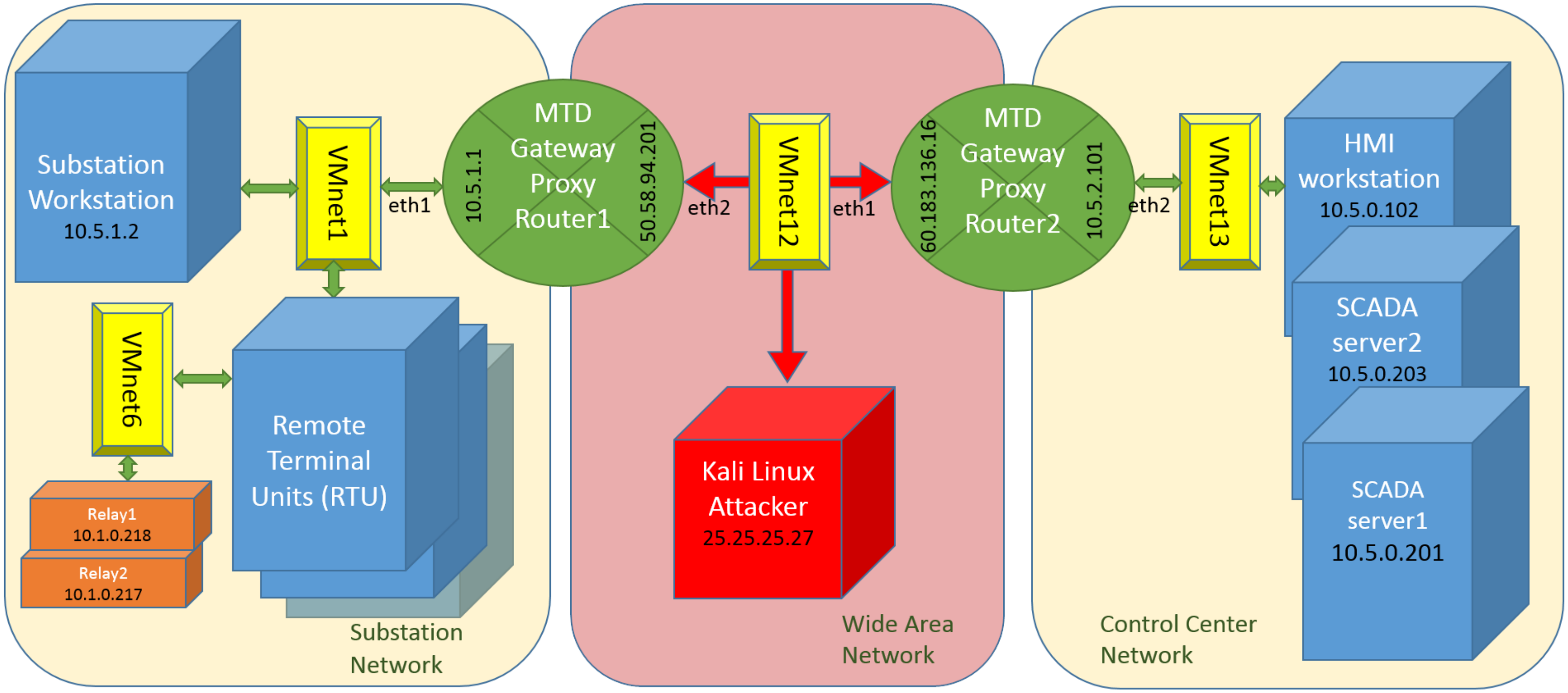


Figure 6. *Architecture of experimental setup for MTD IP hopping*

The probability of successful attack decreases with increasing rate of moving target dynamics, like the IP hop rate, subnet IP address range, etc., thereby, increasing the attack complexity and cost by decreasing the attack surface.

3.3. Algorithm Design and Implementation

The proposed MTD algorithm is deployed only on the Gateway proxy routers on the Control Center and Substation network. This ensures that the end user SCADA applications transparent to the MTD technique. The routers are VMs running Debian Linux distribution and the IP mutation is carried out by programming the Linux kernel routing table and also configuring the networking interfaces to dynamically change their configurations as defined and orchestrated by the IP Hopping algorithm.

3.3.1. IP Hopping Algorithm

Figure 11 shows the proposed IP hopping algorithm that has been implemented in the gateway routers. Figure 11, gives a very simple flow graph depiction of the sequence of steps synchronously executed by both the routers. The synchronization is established by a 3-way handshake protocol as shown in Figure 7.

The MTD1 node on the left hand side represents the gateway router on the Substation network and the MTD2 node represents the gateway router on the Control Center network. As soon as the MTD algorithm is activated on both the routers, MTD1 generates two seeds (random numbers), and MTD2 starts to listen to the network channel to hear from MTD1 for the newly generated seeds.

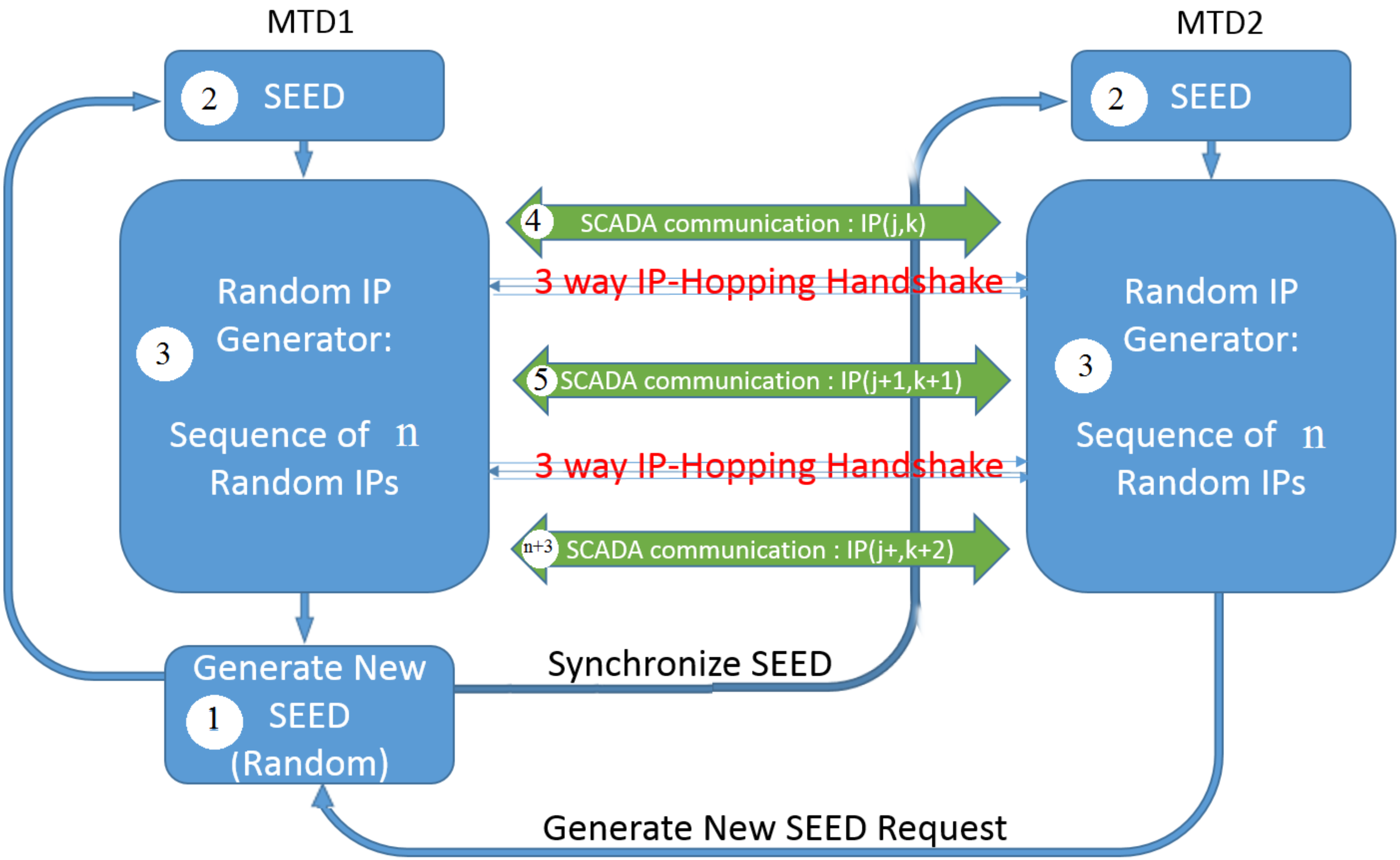

MTD1
MTD2
2 SEED
2 SEED
Random IP Generator:
3
Sequence of n Random IPs
Random IP Generator:
3
Sequence of n Random IPs
4 SCADA communication : IP(j,k)
3 way IP-Hopping Handshake
5 SCADA communication : IP(j+1,k+1)
3 way IP-Hopping Handshake
n+3 SCADA communication : IP(j+,k+2)
Generate New
1 SEED
(Random)
Synchronize SEED
Generate New SEED Request

Figure 7. *Proposed IP Hopping Algorithm*

We assume that the system is initially free from any attacks before activating MTD. Then, both the routers start to generate two sets (j, k) of n Random IPs by using the random IP generator function in combination with the seeds as the initial vector.

The random IP generator is a function which generates valid random IPs based on the Internet Assigned Numbers Authority (IANA) standards of available IPv4 address blocks within the subnet. The gateway routers here assume the availability of an entire /24 subnet of the IPv4 address from the ISP and hops between the 254 (2^8=256 entire range - 1broadcast address – 1network address). It can also be assumed to have multiple Public IP addresses (let’s say ‘m’ Public IP address range of /24 subnet each), thereby providing network route mutation resulting in increased path obfuscation and proactive defense to traffic attacks. This assumption gives us a total range of {m*254} IP address space to be used for each of the MTD routers for the defense strategy.

The sequence of random IPs generated by the random IP generator function depends on the initial seed and is reproducible. Hence, this preserves the order synchrony between the two gateway routers. Figure 3 shows a timing diagram of how the routers forward the SCADA communications using the list of generated sequence of Random IPs at their external interfaces during different time instances.

Thus, at any instance, for an iteration ‘i’, MTD1 and MTD2 will have IP_{j+i} and IP_{j+i} as their external interface IPs. Since only both the routers have knowledge of the two sets (j, k) of the IP addresses, the routers are safe from being targeted. After traversing through the entire sequence list of randomly generated IPs, MTD2 initiates a Request to MTD1 to generate a new seed value.

Thus, both the routers once again iterate through the handshake procedure with different values of Seed for the new iteration, thereby, resulting in different sequences of randomly generated IPs. In this algorithm, the seed is the secret which establishes different levels of randomness in the mutated IP values of the external interface of the gateway routers. Hence, the seed should be kept secret to prevent the attacker from guessing it and therefore it is transmitted to MTD2 by using Public Key Infrastructure (PKI), locally setup in the gateway routers.

3.3.2 3-way IP-Hopping Handshake

The timing diagram in Figure 3 shows how the 3-way handshake: MTD1SyncTx, MTD2SyncTx and an Ack are used to establish synchronization for the orderly mutation of the IPs between the MTD routers and preventing packet losses through timely connection establishment. This helps the routers to hop from one IP address set ($IP_{j,k}$) to the next IP address ($IP_{j+1, k+1}$) in the sequence determined by the IP Hopping algorithm. The 3-way handshake is always initiated by MTD2. After waiting for the hopping interval timer (4s), it transmits a SyncTx packet to MTD1. MTD1 receives the SyncTx from MTD2 and sends another SyncTx back to MTD2. With successful exchange of SyncTx, MTD2 sends an Ack completing the 3 way handshake.

In case of the constant rate MTD, MTD2 reinitiates the 3-way handshake after every fixed hopping interval timer (4s), while the variable rate MTD uses random hopping interval timers which are uniformly distributed within the permissible lower (3.14s) and upper threshold (6.47s) values of the hopping interval as evaluated and defined by the performance of SCADA systems and the defense of the communication network.

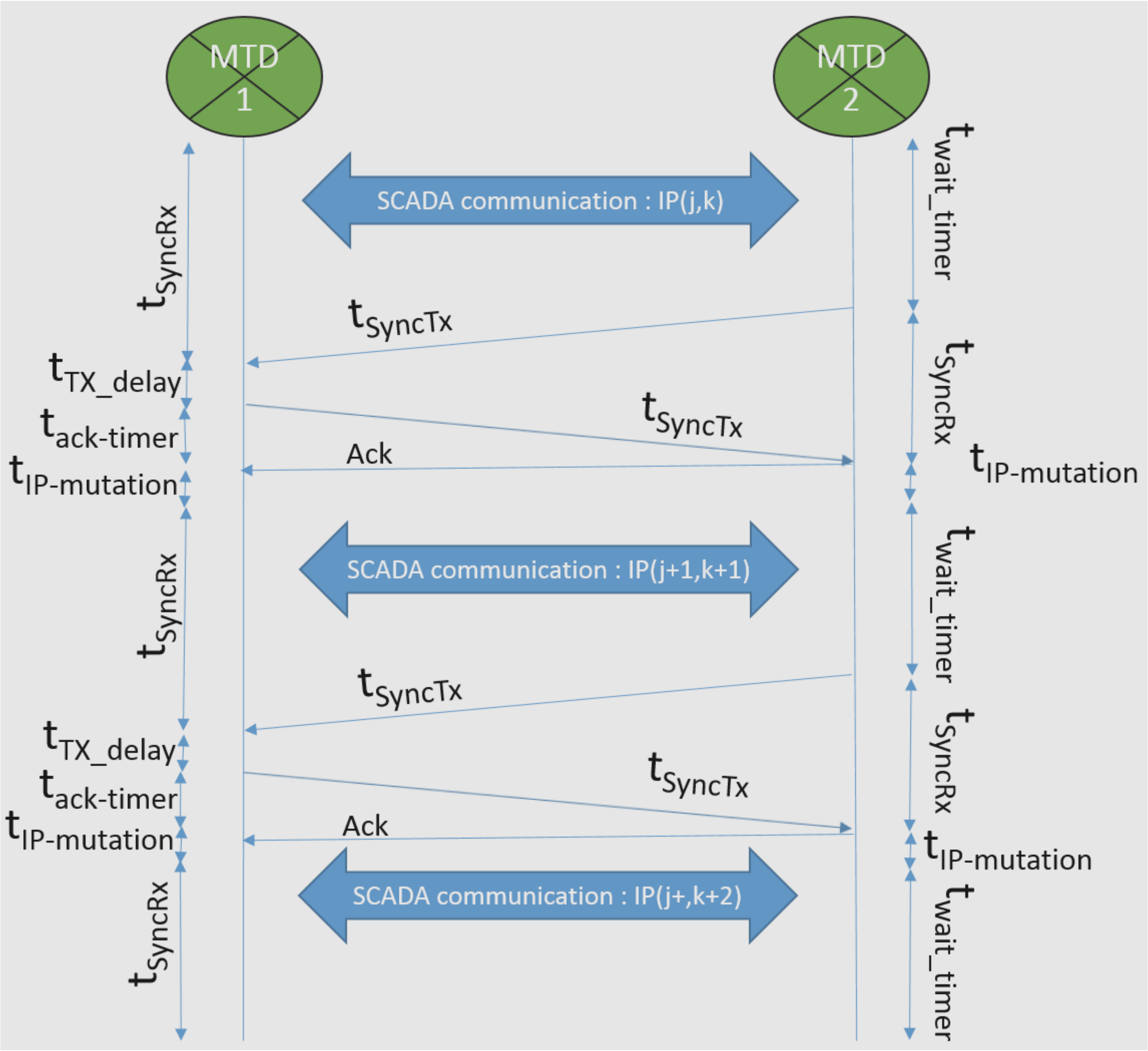


Figure 8. *3-way IP-Hopping Handshake & timing diagram*

The timing parameters like the t_{wait_timer}, t_{SyncRx}, $t_{IP\text{-}mutation}$, t_{TX_delay}, $t_{ack\text{-}timer}$ are the internal process timers running on the end nodes which are also defined by evaluating the throughput and delay characteristics of the communication channel to ensure uninterrupted connection service, and the t_{SyncTx} is the transit time taken by the SyncTx packets to reach the destination router nodes.

CHAPTER 4

IMPLEMENTATION AND EXPERIMENTAL EVALUATION

This chapter discusses in detail the various possible ways of implementing a Network-based MTD in a SCADA environment. The implementation and experimentation is evaluated in the Cyber-Physical Security Testbed available in the PowerCyber Lab in Iowa State University[13]. The testbed architecture is shown in the figure 9.

4.1 Implementation methods

In this section, four different ways of implementing a network-based MTD is discussed although there could be more other possible methods. The first method shows how a host address hopping and routing path randomizing MTD can be implemented using SDNs. This one is just a theoretical proposal and is not implemented in the PowerCyber testbed, whereas the remaining three methods are implemented in the testbed, out of which the one with Gateway Proxy is used for further experimentation and evaluations.

4.1.1 Implementation using SDNs:

Software Defined Networking (SDN) is a newly advancing networking prototype, which provides features to manage and control network configurations & services (like load balancing, optimizing QoS, traffic engineering, fault tolerance & system resiliency, etc.) and resources from a central control point.

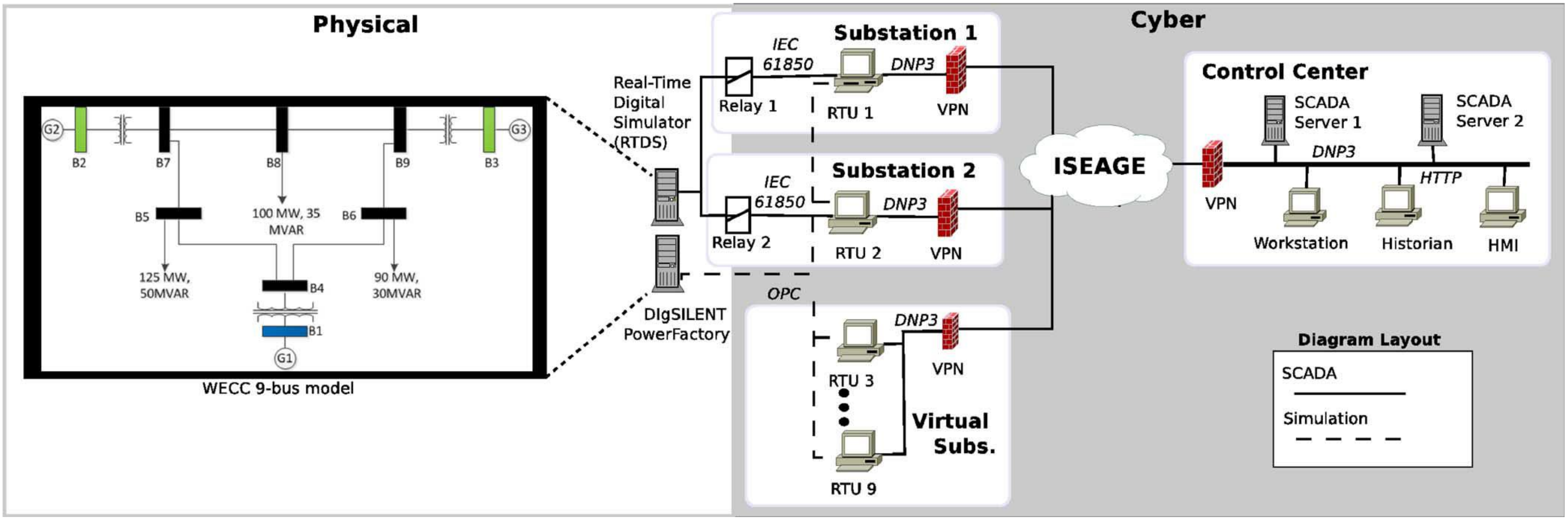


Figure 9. *Power Cyber Testbed architecture with SCADA protocols*

SDN aims at achieving a centralized network control by breaking the vertical integration and extracting the networks control logic from the underlying forwarding devices (switches). Especially the key feature of SDN is to provide dynamic network programmability (changing routing flows) allowing them to have unparalleled flexibility in reconfiguring an IP network in runtime.

The smart grid SCADA communication infrastructure can adopt SDN technology to add more resiliency against accidental faults and directed malicious attacks. Conventional smart grid communication networks involve fixed and non-adaptive networking functionality, and therefore changing the configuration at normal operation is difficult, tedious and cumbersome. This impedes the reconfiguration at faulty events either due to general failure or malicious attacks. Also this creates a performance bottleneck to modern grid infrastructure which involves high bandwidth demanding devices like Phasor Measurement Units (PMUs) and Advanced Metering Infrastructure (AMI).

With programmability, network operators can redefine the existing smart grids to dynamically adapt to changes, minimizing fault times, with quicker self-healing capabilities. Recovery techniques involve introducing redundant fault paths, and more easily manageable ways to isolate fault locations to resist the severity of attack propagation [20]. Also power system specific applications can be built on the controller’s APIs which are more focussed on mitigating attacks which are directed on smart grid applications (like EMS, AGC, State Estimation etc.) by looking into specific details of the packet transmission patterns (like analysing inter-arrival time for DNP3, IEC 61850 GOOSE packets) which have more predictable characteristics in SCADA environments.

Especially SDNs find significant role in network based Moving Target Defence strategies by rapidly increasing the attack cost [21]. With a global view of the entire underlying network architecture, it can be more appropriately used for early-detection of specific attacks like Distributed Denial of Service (DDoS) attacks [22], and then immediately upon detection it can logically redefine the network topology by modifying the routes to preserve the quality of grid control. Also one can employ proactive defence techniques like host IP mutation [22] to prevent outside attackers from enumerating the network topology and OS fingerprinting.

Along with other detection techniques like Intrusion Detection System IDS, it can prolong the attack time, by increasing the frequency of IP address randomization at once it receives the alarm trigger. Such approaches shrink the time window for an adversary to exploit further vulnerabilities.

SDNs also has wide scope for Researcher and Students, as they provide a great experimental platform to design and evaluate cyber-attack/defence use-case scenarios in SCADA environment. This is possible with the flexible and scalable nature of SDNs to redesign the network architecture facilitating uninterrupted experimentation without involving significant changes. This motivates the testbed development and federation among different universities and national labs.

Isolated topologies can be created which promotes simultaneous use of multiple experiments sharing the limited resources with no compromise in experimentation quality, thereby promoting a classroom lab environment for students to increase the comprehension of cyber security for smart grids.

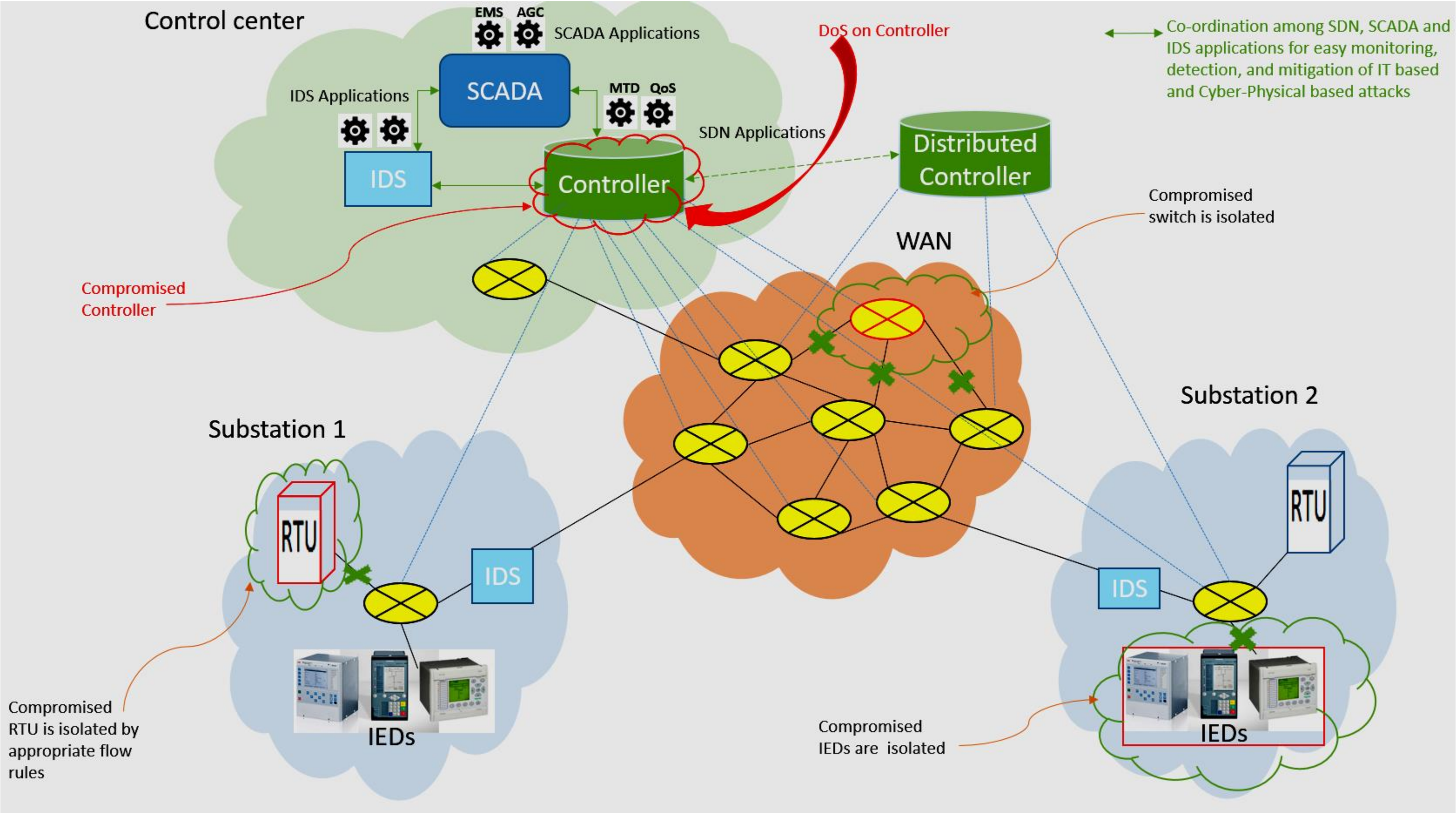

Control center
EMS
AGC
SCADA Applications
SCADA
IDS Applications
MTD
QoS
SDN Applications
IDS
Controller
DoS on Controller
Co-ordination among SDN, SCADA and IDS applications for easy monitoring, detection, and mitigation of IT based and Cyber-Physical based attacks
Distributed Controller
Compromised switch is isolated
WAN
Compromised Controller
Substation 1
Substation 2
RTU
IDS
IEDs
RTU
IDS
IEDs
Compromised RTU is isolated by appropriate flow rules
Compromised IEDs are isolated

Figure 10. *Network-based MTD implementation design using SDNs*

However, despite the above advantages of SDNs, it also suffers from some of the severe drawbacks. The separate control and data plane proves to be a threat. An attack can be more dangerous if an adversary compromises the controller. A malicious network reconfiguration control from the compromised controller can destruct the entire infrastructure. Hence the controller must be highly attack resilient. But it is impossible to expect a controller to have zero vulnerabilities. Also the controller is a single point of failure, and a DoS on a controller will have a huge impact. One possible solution is to have distributed controllers. This makes SDN more viable for smart-grid Research environments rather than the real time operation environment.

4.1.2 Implementation using NAT:

Network Address Translators are often enabled within a gateway device (like modem, router or firewall, etc.) and can be a part of the local system itself. The proposed IP hopping MTD mechanism which will be discussed in detail in the later chapters are implemented using the following three techniques namely, using NAT, local scripts and gateway proxies, out of which the gateway proxy was considered for experimental evaluation purposes.

Since, the Linux systems comes with iptable/ipchain firewall within them, they are used for the implementation of address translation as well as dynamic filtering of old IP addresses. The main purpose of a NAT is to translates public (WAN) IP address range to private (within a corporate LAN) IP address range (defined by RFC 1918), to ensure their conservation of IPv4 address space (as IPv4 address spaces are depleted with more number of internetworking devices), but it also gives an add-on security feature, as shown below in the figure 11.

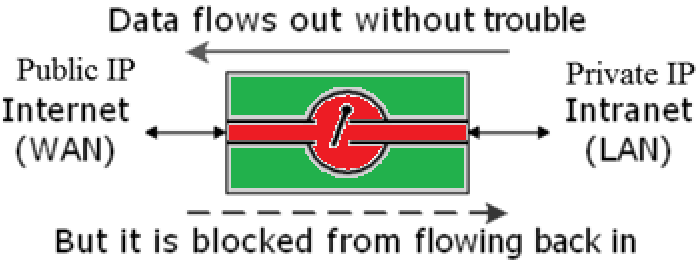

Data flows out without trouble
Public IP
Internet
(WAN)
Private IP
Intranet
(LAN)
But it is blocked from flowing back in

Figure 11. *Inherent security of NAT*

Thus, it serves like a one-way valve, blocking all the other unauthenticated traffic initiated from the WAN to the LAN. But still, as we previously discussed, an attacker who could spoof or masquerade an authenticated existing traffic session with static translated IP address, he could penetrate through this and execute a successful attack. Therefore, the proposed MTD scheme uses a dynamic algorithm to change the translation IP address every few seconds, with refreshing the Conntrack (connection tracking) entries and flushing the connection states to make it stateless for uninterrupted session migrations. The SNAT and DNAT connection entries are recreated with each new translation, which are virtual states when the source address or destination address has been altered respectively by the connection tracking system.

The figure 12 shows the internal process and flow diagram of iptable in the Linux machines. It has 4 separate process with rule entries in their respective tables (Mangle, NAT and filter). The routing process itself has three other sub-functions namely,

1. Pre-routing: ingress traffic routing
2. Forwarding: just forward packets to the local host
3. Dropping/Blocking: two different methods to filter packets with and without response
4. Post-routing: egress traffic routing

The MTD algorithm is implemented at either the Pre-routing or Post-routing NAT table as shown in the figure below, indicated by the blue spot.

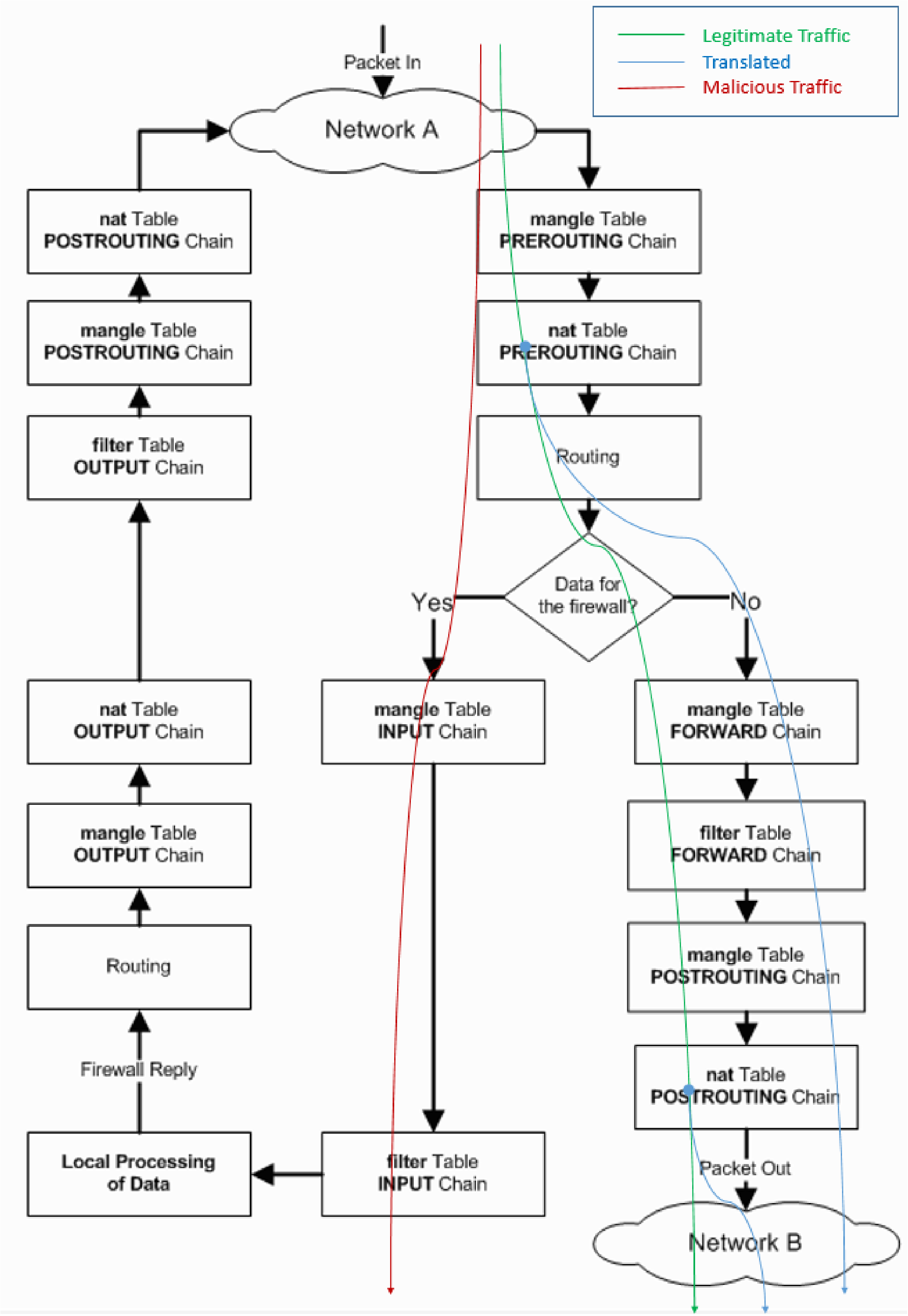
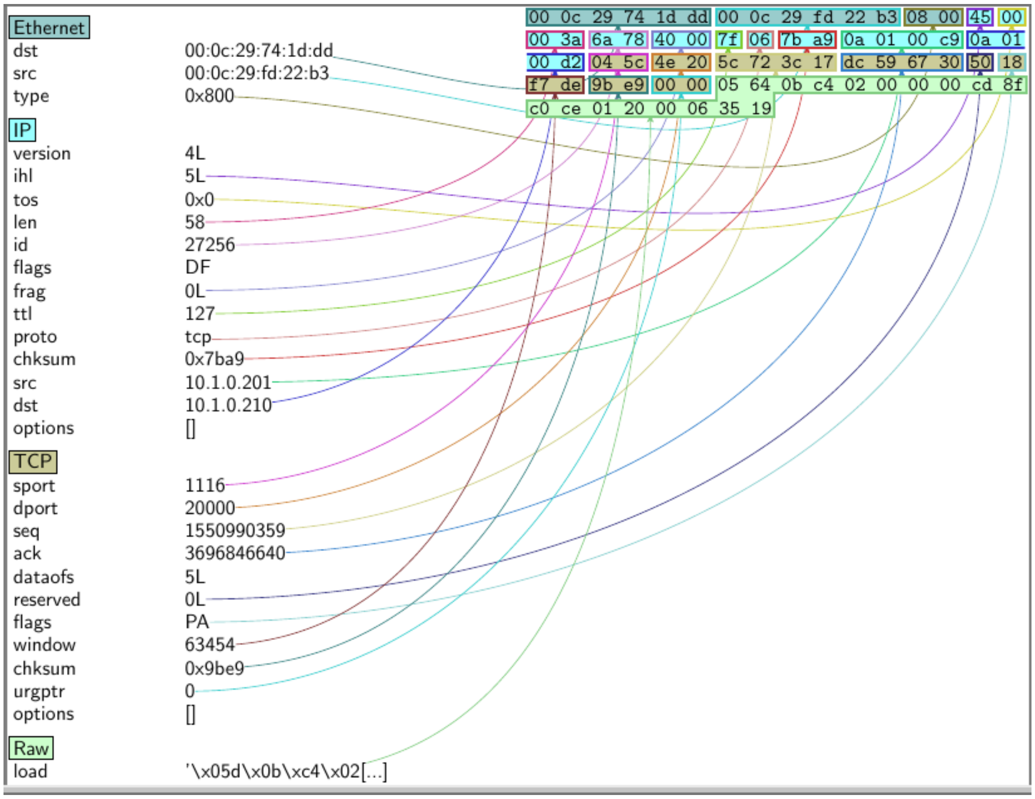

Legitimate Traffic
Translated
Malicious Traffic
Packet In
Network A
nat Table POSTROUTING Chain
mangle Table POSTROUTING Chain
filter Table OUTPUT Chain
nat Table OUTPUT Chain
mangle Table OUTPUT Chain
Routing
Firewall Reply
Local Processing of Data
mangle Table PREROUTING Chain
nat Table PREROUTING Chain
Routing
Data for the firewall?
Yes
No
mangle Table INPUT Chain
filter Table INPUT Chain
mangle Table FORWARD Chain
filter Table FORWARD Chain
mangle Table POSTROUTING Chain
nat Table POSTROUTING Chain
Packet Out
Network B

Figure 12. *Process flow diagram of iptable with legitimate and malicious traffic*

From the figure 12, we can see that the legitimate traffic, which has the updated IP parameters, from the other end gateway NAT satisfies the NAT routing chain rules and bypasses the firewall to go to the internal network. Whereas the replay or spoofed malicious traffic is filtered by the updated firewall rules, which is incongruence with the changes in the NAT by the MTD.

4.1.3 Implementation using local scripts:

This can be a slow and computationally inefficient technique which uses local application scripts for implementing the MTD algorithm. But for less traffic scenarios, this can be highly flexible, simple, easy to implement and interoperable. This implementation makes use of python scripts built using one of its library extensions called Scapy.



```
Ethernet
dst        00:0c:29:74:1d:dd
src        00:0c:29:fd:22:b3
type       0x800
IP
version    4L
ihl        5L
tos        0x0
len        58
id         27256
flags      DF
frag       0L
ttl        127
proto      tcp
chksum     0x7ba9
src        10.1.0.201
dst        10.1.0.210
options    []
TCP
sport      1116
dport      20000
seq        1550990359
ack        3696846640
dataofs    5L
reserved   0L
flags      PA
window     63454
chksum     0x9be9
urgptr     0
options    []
Raw
load       '\x05d\x0b\xc4\x02[...]

00 0c 29 74 1d dd 00 0c 29 fd 22 b3 08 00 45 00
00 3a 6a 78 40 00 7f 06 7b a9 0a 01 00 c9 0a 01
00 d2 04 5c 4e 20 5c 72 3c 17 dc 59 67 30 50 18
f7 de 9b e9 00 00 05 64 0b c4 02 00 00 00 cd 8f
c0 ce 01 20 00 06 35 19
```

Figure 13. *Analysing DNP3 packet fields with Scapy protocol dissector*

Scapy is a packet manipulation tool, which can sniff packets from the NIC card of the PC and modify packets on fly, and retransmit them through the same or other Network adapters available in the system. The figure 13 above shows the dissection of a DNP3 packet captured on fly, and the fields analyzed by the Scapy tool. It has an inbuilt dissector to point to the protocols fields, as designed by a template. IP.src here points to source IP address of the packet 10.1.0.201 and IP.dst here points to the destination IP address of the packet 10.1.0.210.

The MTD IP-Hopping algorithm is interfaced through the Scapy packet manipulator, to change the IP address dynamically, and recalculating other header fields like the new IP header checksum, and then retransmitting it into the internal network. Thus effectively it does the same function as previously performed by the NAT, but here it is system independent and processed locally rather than at the kernel level. This saves the resources of the gateway systems to be used more effectively.

4.1.4 Implementation using Gateway Proxies:

The proxy gateways can be either on dedicated hardware or as a software program running on the local host machine. Here, in this thesis, a dedicated VM running Linux is used for running the proxy gateway, which helps switching from one network to another (here from the Substation or Control Center network to the WAN and vice versa). The figure 9 shows the use of Proxy gateways used by the substations and control centers to communicate via the WAN.

The remainder of this thesis will describe, how the IP-Hopping MTD is implemented in this method and evaluated considering various security parameters. As shown in the figure 14, the proposed MTD mechanism helps in dropping the flooding traffic injected by the attacker as well other unauthenticated or spoofed targeted attacks. This type of implementation is the easiest one compared to the previously discussed techniques and therefore this thesis uses it for further analysis and testing, although for more realistic use cases one might need to consider the previous implementations.

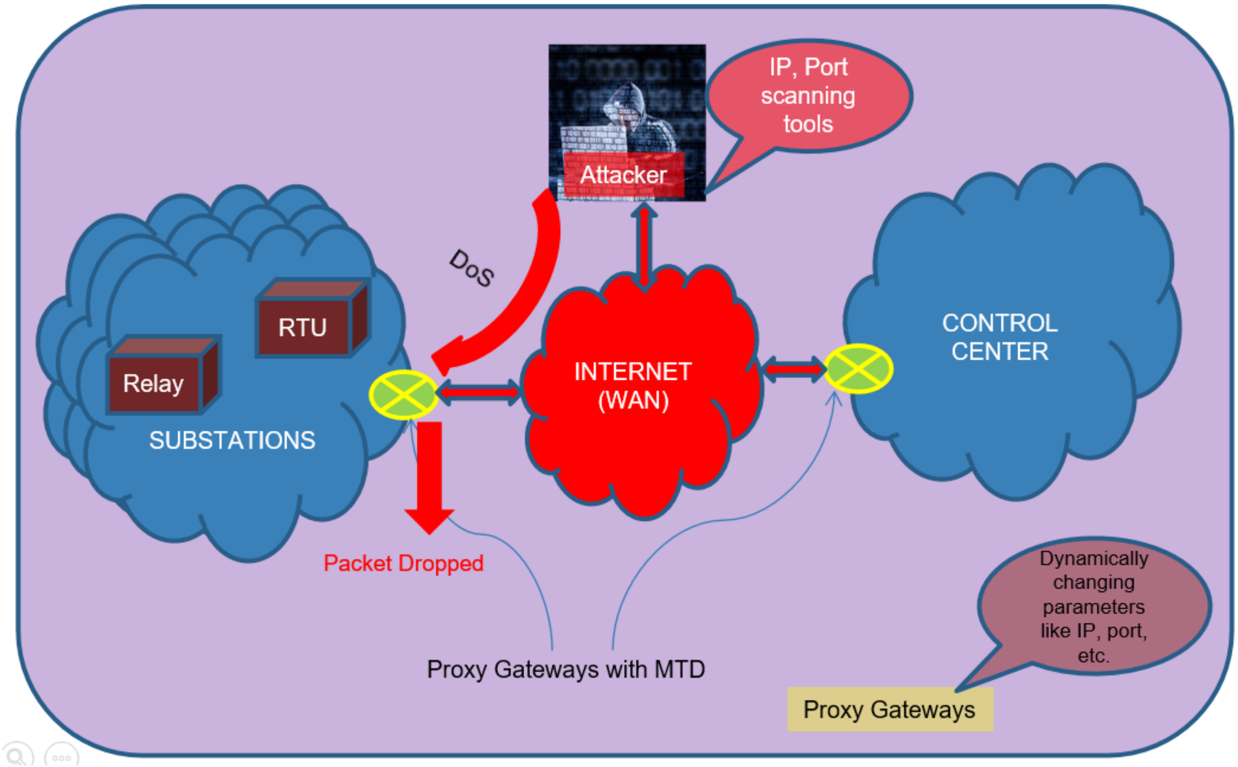


Figure 14. *Topology of Proxy Gateway based network MTD*

4.2. Analyzing Throughput & Delay characteristics:

It is very much essential to evaluate the performance metrics like throughput and delay timing parameters of the end-to-end SCADA communication as the control applications are time sensitive and could have huge impacts due undesirable delay or jitter introduced by the MTD mechanism.

Table 3. *Delay overhead introduced by MTD*

RTT mean (ms)	Without MTD	Hopping interval of Constant rate MTD					Variable rate MTD
		3	4	5	6	7	
	48.26	2478.44	50.63	50.48	50.43	50.42	50.59

Table 3 presents the Round Trip Time (RTT) in ms for different cases as obtained during experimentation. For the purpose of evaluation, the DNP3 session is terminated if the DNP Keep–alive timer expires. In our case studies, the SCADA servers were configured with DNP connect-timeout = 2000ms.

When the MTD routers hop IP address once every 3 seconds, because of the transient flushing of the router's routing cache and network interface configuration changes, the response time of the system increased. This along with the TCP retransmission timeouts, delayed the forwarding of packets in the network queues, resulting in a mean Round Trip Time (RTT) of 2478.44ms leading to termination of the DNP3 sessions due to timeouts. Observing the mean RTT for hopping intervals of 4s and above, we identify that the delay introduced (approx. 2.23ms) is acceptable to the SCADA system as it did not affect the overall communication at the application level.

Figure 15 shows the throughput characteristics of the constant rate MTD with various IP hopping rates. The color dots show the individual packet throughput at different time instances The dips in the throughput data point to the instances at which the router mutates its external interface's IP address, resulting in some packet losses, which increase with increasing hopping rate. But here the throughput is calculated at the network layer by counting the packet size of all the TCP packets (TCP session management packets as well the TCP packets encapsulating the

DNP packets) that are received by the Network Interface Card, whereas the RTT delay is measured at the SCADA application layer that only looks out for successful DNP3 packets that are accepted by the Power TG application.

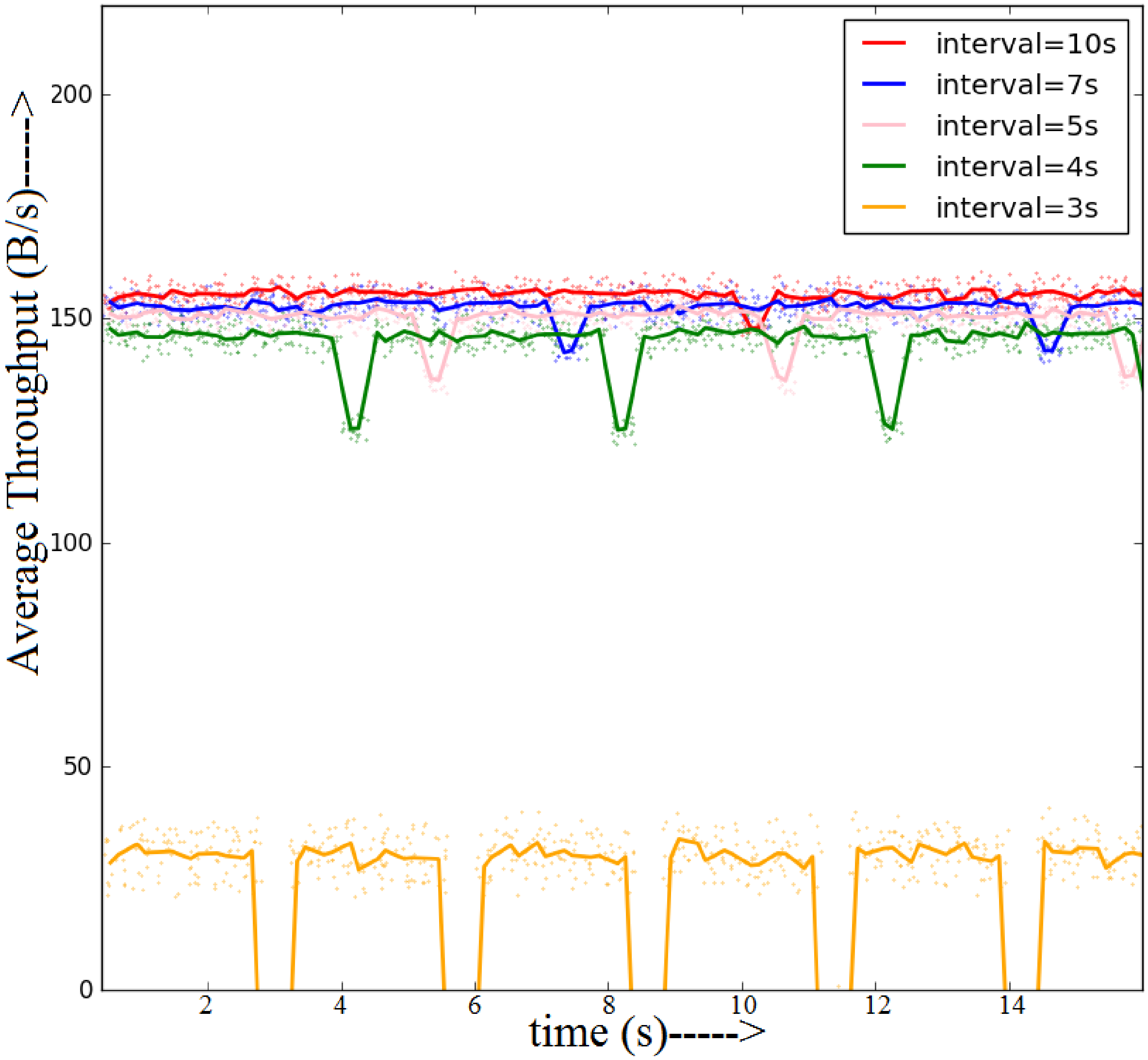


Figure 15. *Average Throughput vs. Time*

4.3. Choosing the best hopping interval

Determining the best hopping rate for a SCADA environment is crucial to the efficient functioning of the proposed MTD mechanism. It must not only provide better security, but more importantly, it should retain the SCADA applications' communication sessions, providing availability through authorized access to the end devices in the loop.

In order to evaluate this we measure the attacker's probability to execute a successful attack as P_A. We use the attack scripts models used in [11] and [12], which were modeled for the SCADA testbed to act as a MitM and inject malicious command and control. The successful attack probability P_A intuitively captures a number of factors such as the total IP address range available for MTD, no. of attack scanning probes, no. of vulnerable hosts and the IP-Hopping rate.

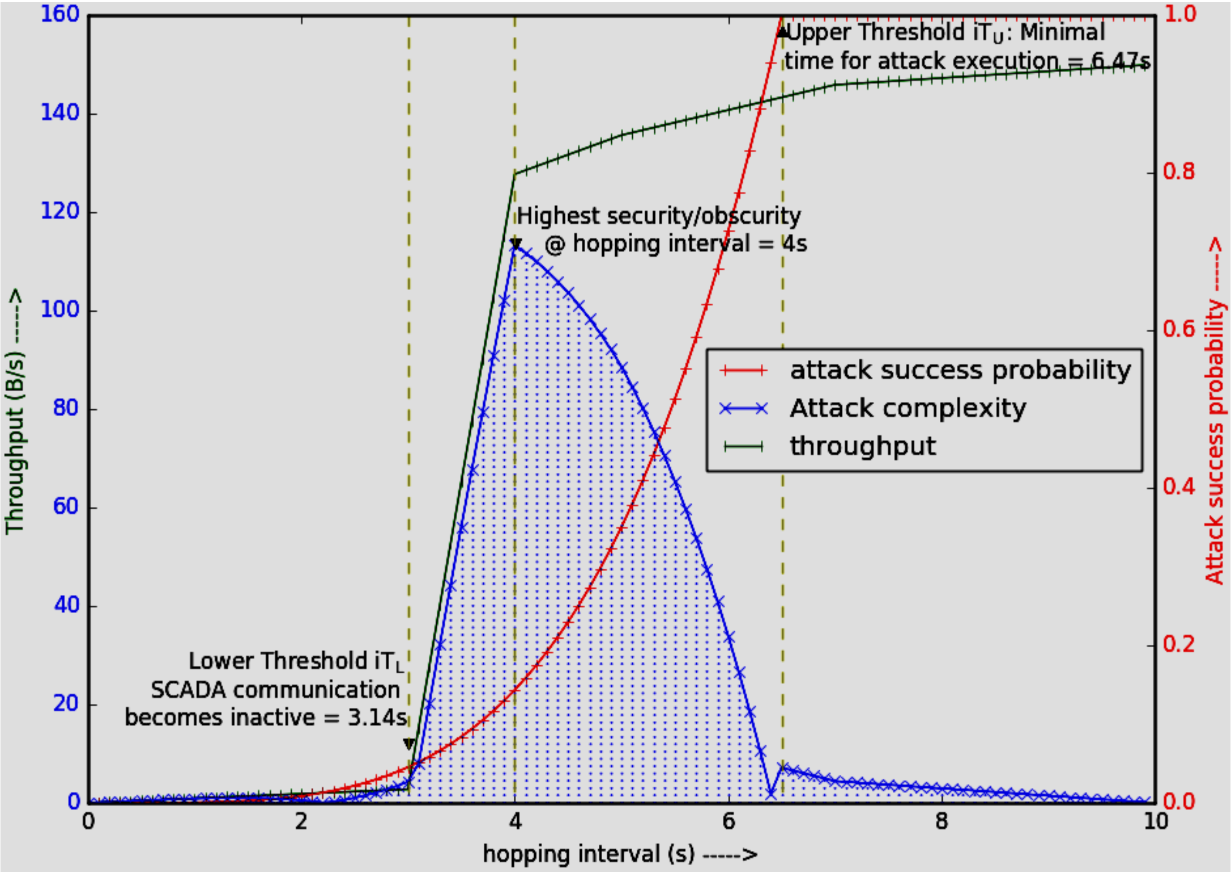


***Figure 16**. Selecting the best IP hopping interval for the SCADA network*

In order to target a particular device in the SCADA network the attacker has to identify the gateway router's external interface's active IP address configuration. Hence, here we assume the attacker to use Nmap network host discovery service [7] to scan for active IP address in the network, and we find that the attacker takes approximately 25ms to scan and discover the status of a single IP.

Considering the worst case scenario of defense, where the MTD uses only a single subnet range (with only 254 addresses) for mutation, the entire host range is scanned successfully in 6.47s, resulting in an Upper bound constraint iT_U for the hopping interval. Also, the SCADA communication sessions are terminated as the hopping interval goes below 3.14s posing a lower bound constraint iT_L. The scanning results are plotted in Figure 16 along with the throughput to determine the best hopping rate. The best lowest hopping rate is one that provides the optimal tradeoff between performance as well as defense, for the worst case scenario.

4.4 Considering a more realistic WAN with multiple hops:

The previous testbed setup had only a single path and single hop in the WAN, but in reality a WAN has multiple paths and multiple hops seperating the Control Center network and the Substation network. Therefore, to evaluate a minimal realistic scenario, we expand the single network into a 10 network with 2-hop (shortest path) or 4-hop (longest path), and 12 different routes or path, as shown in the figure 17 below.

Considering this scenario, each of the MTD devices are connected to two public interfaces, with additional cost of purchasing another connection, we have better security and performance. As we can see, now the probability of traffic flowing through a single route is reduced by 12. Therefore the probability that an attacker dropping a traffic in any one route is also reduced.

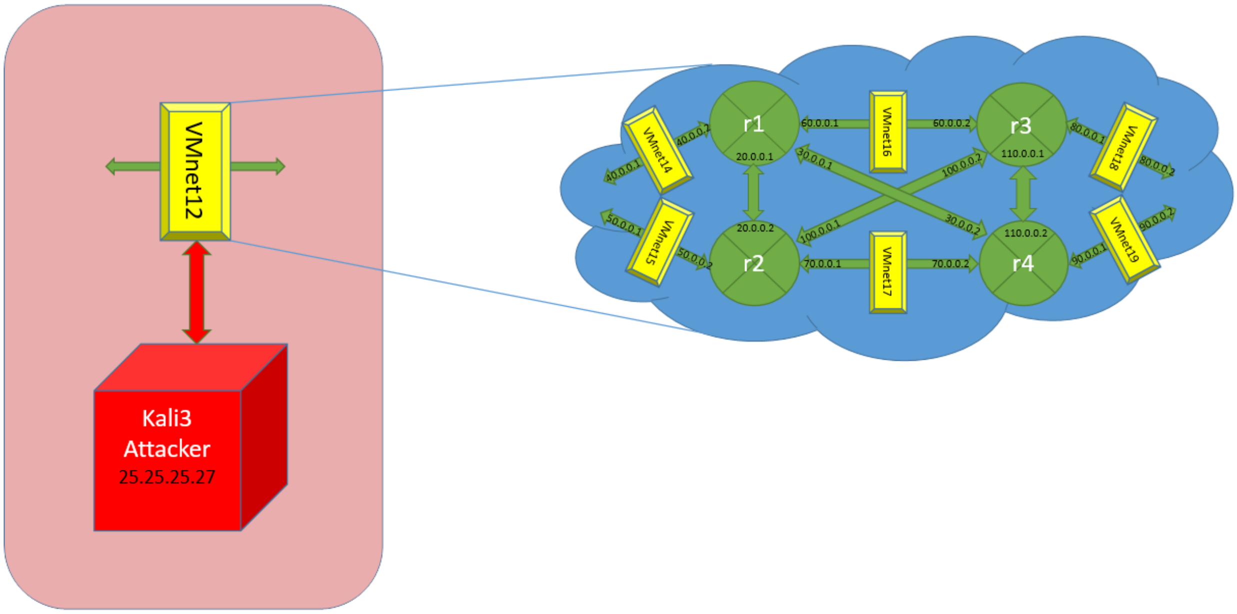


Figure 17. *Expanding the single hop WAN to a multi-hop, multi-path WAN*

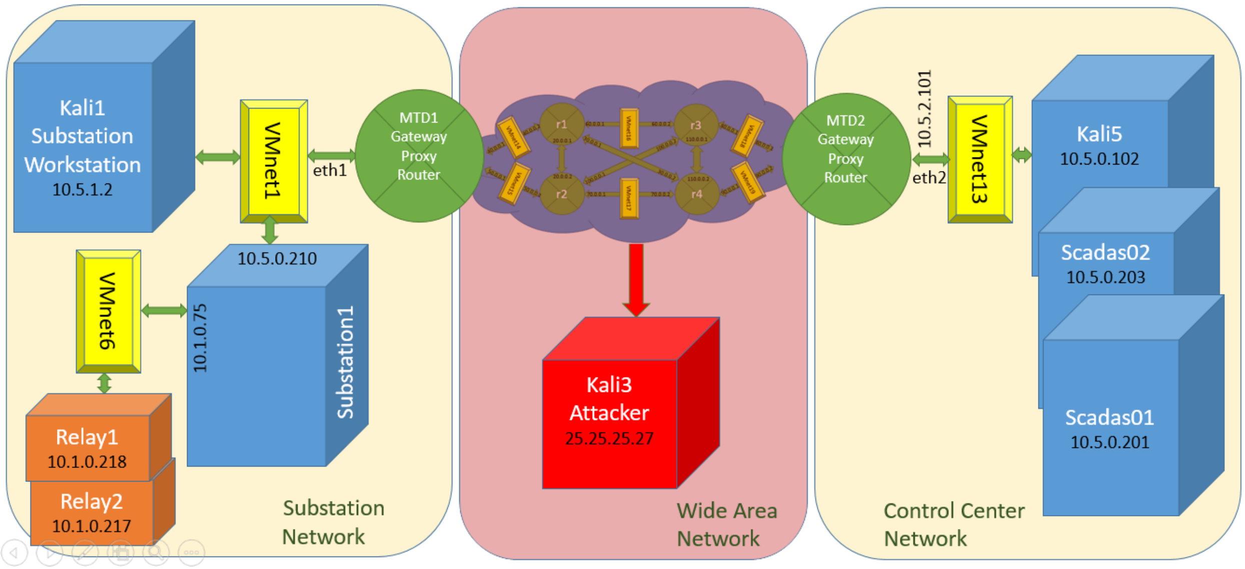


Figure 18. *Updated Topology with a realistic WAN*

At the same time, the hopping rate can also be increased as we have additional connection resources. This can be understood from the figure 19 shown below.

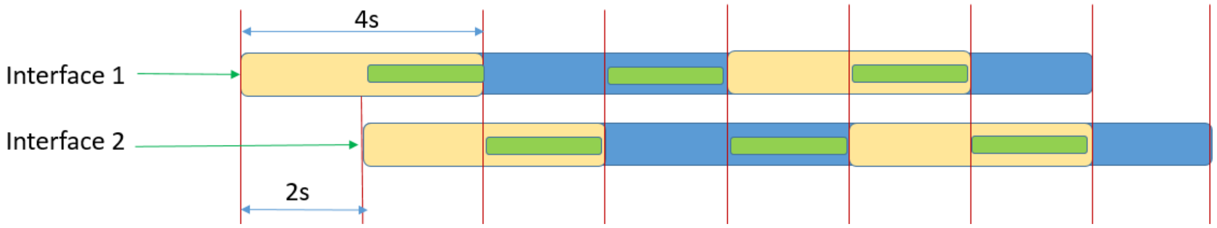


Figure 19. *Doubling the Hopping Rate*

The best hopping rate for a single interface (4 seconds) is used at each interface to balance the SCADA traffic at two different interfaces simultaneously. The green color indicates the interface actively being used for SCADA communication (route switching happens after every 2s, after the interface changes its IP address, so that the disturbance created by IP hopping in the network settles down for better throughput), whereas the yellow and blue color indicates the alternating IP addressess acquired by the individual interfaces. This doubles the hopping rate with better performance and throughput at the expense of additonal cost.

4.5 Choosing the best IP range:

For this purpose, an attack model is needed to evaluate the probability of successful attack. Hence, a malicious command injection (breaker trip open command attack) is considered. The attack is scripted in python and makes use of the following communication sequence for a successful execution of the attack. Therefore, if the defense system is changing its IP address at the previously observed best IP hopping rate with an interval of 4s, then the attacker needs to guess or find out the IP address twice to have the entire attack command operated over the RTU.

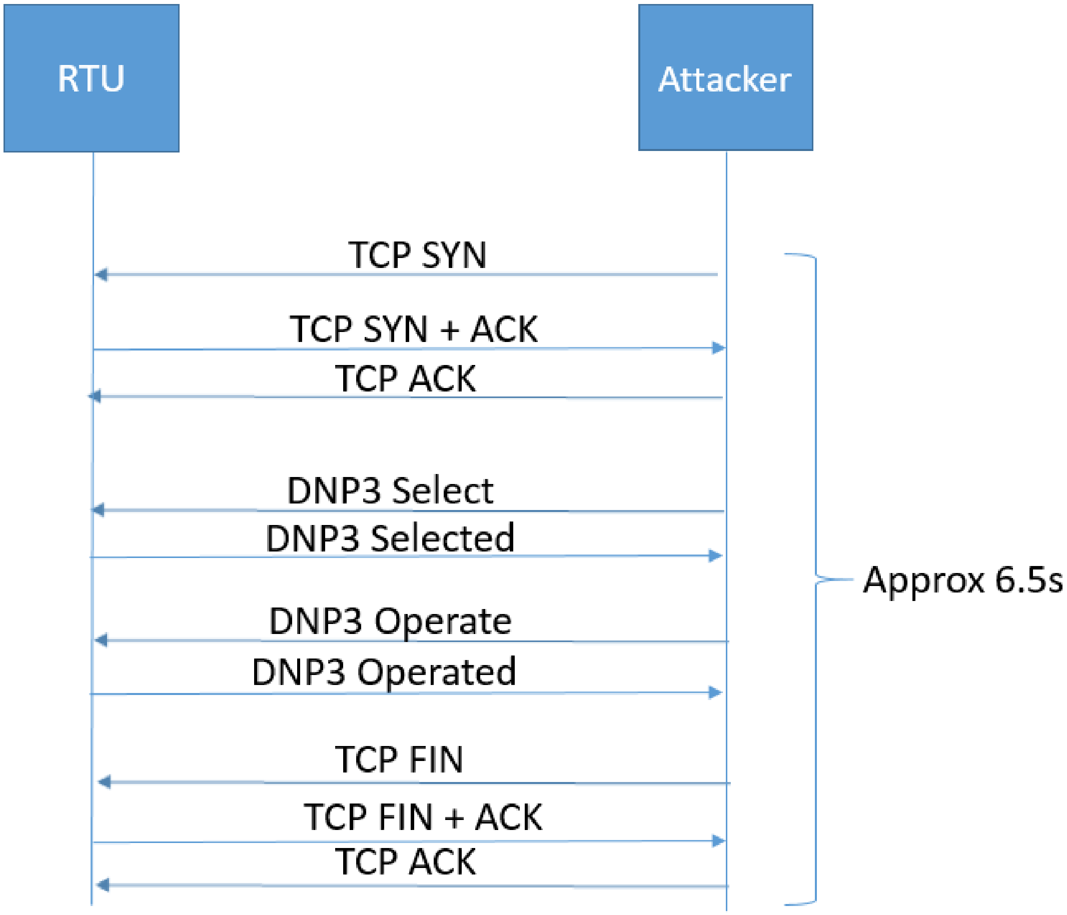


Figure 20. *Malicious Trip Open Attack command packet sequence timing diagram*

Based on this model, the probability of attack success (P_A) is calculated as,

$$P_A = (1/x)^n$$

where x is the no. of IP addresses available in the range, and n is the no. of times slots an attacker needs for a successful attack (to cover 6.5s) .

The figure 21 below shows the Probabliltiy of successful attack for different hopping intervals with respect to increasing IP range. We can see, with less hop interval (i.e., higher hopping rate) the probability of successful attack decreases exponentially.

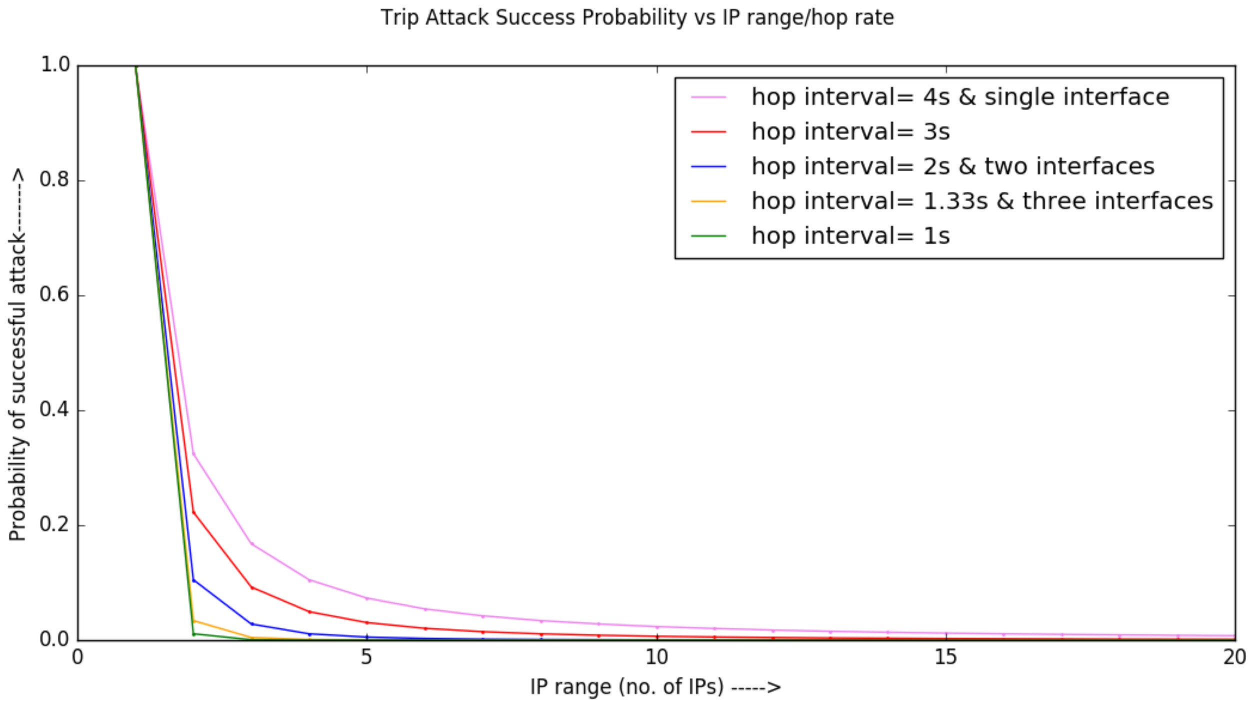


Figure 21. *Choosing best range of IP addresses*

From the above section 4.4, we see that having two interfaces doubles the hopping rate. Therefore with more interfaces, we can achieve faster hopping rates (very less hop interval), helping the defender create more time slots for the attacker to guess (i.e., the attacker will need to predict more within a short time). This increase the attacker cost.

The table 4 shows the range of IP addresses required for different hopping intervals. From the above table we can see that, having more interfaces, not only increases the attackers cost by giving us higher hooping rates, but also less number of required IP addresses for hopping, as the probability of attack success is inversely and exponenetially depend on the no. of available IP addresses and hop rate respectively.

Table 4. *Best range of IP addresses and their corresponding hopping intervals*

Hop Interval (seconds)	**IP range required for P_A = 0.001**	**IP range required for P_A = 0.005**
4	70	107
3	24	33
2	8	11
1.33	4	6
1	3	4

4.6 Other MTD Specific Attack Scenarios and Mitigation

4.6.1 *Scenario1*- Address Range exhaustion attack:

The attacker could try to continuously target any one of the IP address, which he had previously sniffed, to contain the DNP3 SCADA traffic. As the MTD routers keep mutating the IPs from a constant range set, at a certain point in time one of the MTD routers could acquire that IP address which is being targeted by the attacker, thereby leading to a successful execution of the attack.

Mitigation- Considering the most recent attack vector modelled in [11] for the SCADA testbed, the attack is successful only if the targeted IP remains static for 6.5s, as the entire execution of the attack is not atomic but requires establishment of a successful DNP3/TCP session to the substation RTU, and selecting the desired breaker, and executing the operate command. This in total takes approximately 7.5s and if the hopping interval is less than 6.5s, then this attack can be easily prevented, as even in the worst case scenario as described in section 5.2, this condition holds good. But for more sophisticated attacks with less attack execution time, we have to similarly leverage the hardware and processing capabilities of the MTD routers to provide better response, effectively allowing the use of lower IP hopping intervals without affecting the throughput and delay characteristics of the SCADA communication sessions.

4.6.2 *Scenario2*- Traffic analysis attack:

In the above proposed MTD technique, the IP-Hopping algorithm makes use of a fixed hopping interval, which poses as a threat from an attacker who performs traffic analysis, as the time signature of SCADA application sessions are distinct from traditional IT applications, and also are easily predictable. In our case, just by observing the throughput characteristics of the MTD routers traffic, one can easily see that the dips occur at fixed intervals. Figure 6 shows a traffic attack, which uses high spikes of traffic volume only at the hopping instances, so that it is sufficient enough to overload the channel bandwidth to drop or delay the 3-way handshakes, thereby affecting the entire SCADA sessions, as depicted by the green traffic in Figure 22.

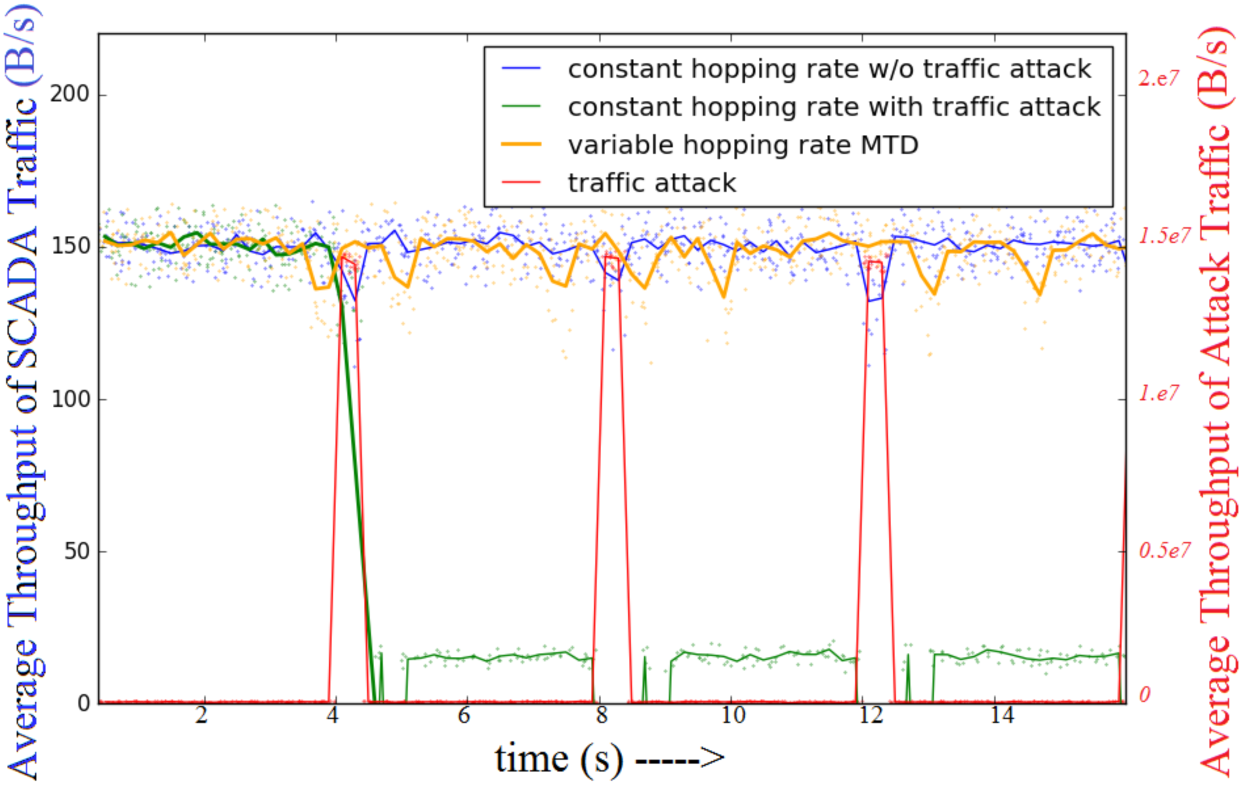


Figure 22. *Constant vs. Variable hopping rate MTD with Traffic Attack*

***Mitigation*- Variable Hopping Rate MTD:**

Instantiating the t_{wait_timer} with a random value bounded between the Upper (iT_L=3.14s) and Lower constraints (iT_U=6.47s), and avoiding handshake exchanges at instances of heavy traffic by sensing the channel can prevent the system from such traffic attack. This also morphs the traffic and makes it obscure for the attackers to comprehend its behavior. Figure 23 shows the histogram of constant and variable rate IP hopping MTD techniques to illustrate this point.

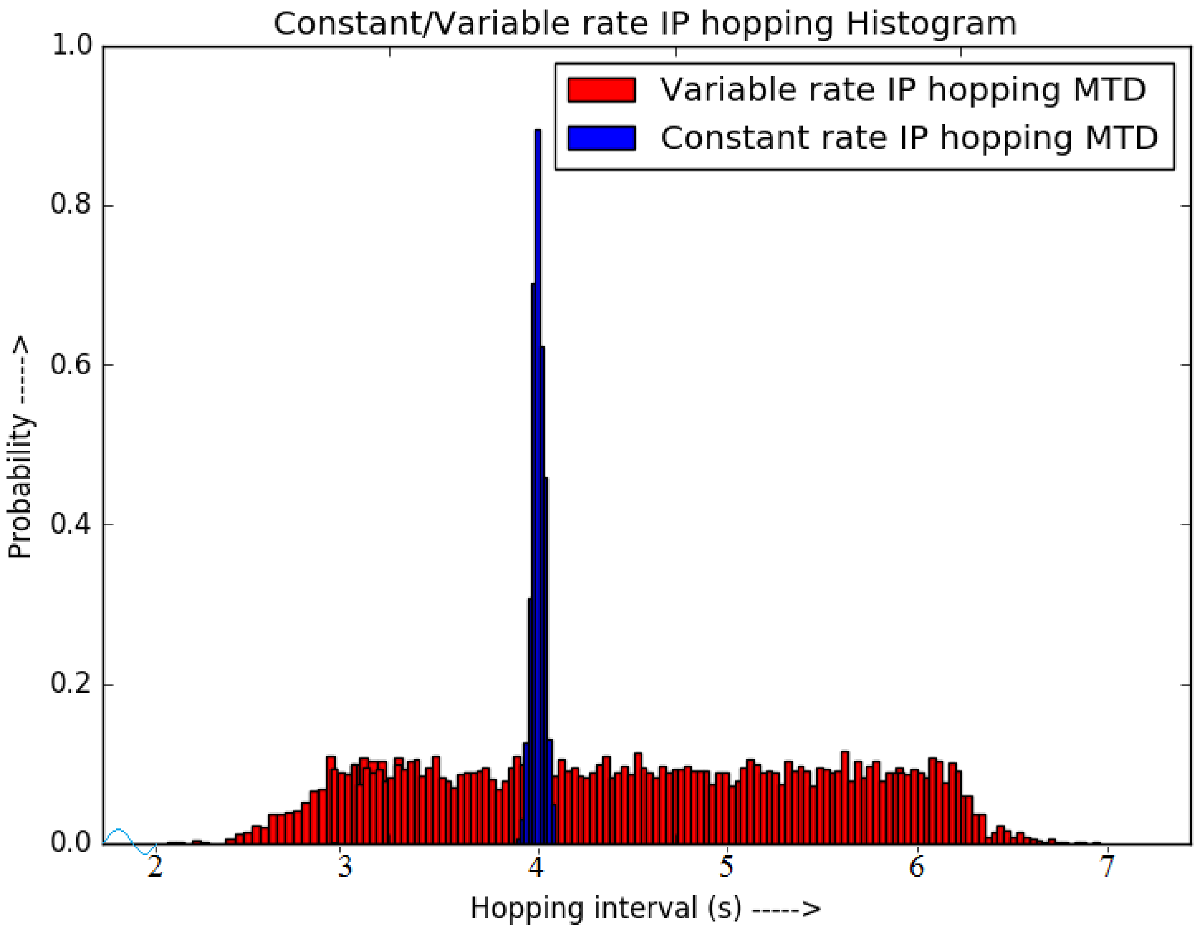


Figure 23. *Histogram of Constant vs. Variable rate MTD*

CHAPTER 5

CONCLUSIONS & FUTURE WORKS

5.1. Conclusions

In this thesis, we highlighted the suitability and feasibility of MTD on a realistic SCADA environment for the power grid to provide increased attack resilience with-out trading off on the SCADA applications' availability. We presented a novel and efficient MTD mechanism using a random and dynamic IP-Hopping technique in conjunction with a 3-way handshake protocol for synchronization among hopping devices. The proposed defense scheme was tested and evaluated on the Power Cyber CPS Security testbed with real-world SCADA software and physical relays.

As part of our experimentation, we varied the IP hop-ping intervals for the proposed MTD algorithm studied its throughput and delay characteristics to determine the best hopping interval considering the worst case scenario into account. Also, we identified certain specific attacks that could exploit the weaknesses of the pro-posed MTD scheme and experimentally evaluated and validated potential mitigation solutions for such at-tacks.

Although the proposed MTD algorithm is simplistic and intuitive, the main motive of this work is to provide a concrete proof-of-concept that MTD techniques can be effectively leveraged in hardening the security and improving proactive defense of SCADA systems that monitor and control CPS such as the power grid without any adverse impacts on the control operations and system availability.

5.2. Future Works

As a part of future work, the following tasks would be performed,

- Incorporate network port hopping using dynamic Port Address Translation and NAT as part of the randomization in addition to IP hopping to increase the attack complexity & cost. Such an approach could decrease the attack surface to a considerable extent, even for sophisticated attack scenarios.
- Optimize the IP hopping algorithm further by incorporating modern cryptographic functions.
- Re-study the experiments with all different implementations and compare their performance and security
- Expand the testbed to multiple substation and multiple control centers
- Incorporate other Power system based applications like RAS, AGC, SE, etc and study their performance with and without MTD.
- Develop CPS flavored MTD on a higher layer to evade the special behavioral characteristics of the Power Grid applications.

REFERENCES

[1] M. Atighetchi, P. Pal, F. Webber, and C. Jones, “Adaptive use of network-centric mechanisms in cyber-defense,” *Proceedings of 6th IEEE Int”l Symp. Object-Oriented Real-Time Distributed Computing,* pp. 183–192, 2003.

[2] Y.-B. Luo, B.-S. Wang, and G.-L. Cai, “Analysis of port hopping for proactive cyber defense,” *International Journal of Security and its Applications*, vol. 9, no. 2, pp. 123–134, 2015.

[3] J. H. Jafarian, E. Al-Shaer, and Q. Duan, “Openflow random host mutation: Transparent moving target defense using software defined networking,” *Proceedings of HotSDN workshop at SIGCOMM’12*, pp. 127–132, 2012.

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

[5] E. Al-Shaer, Q. Duan, and J. H. Jafarian, “Random host mutation for moving target defense,” *Proceedings of the 8th International Conference on Security and Privacy in Communication Networks,* pp. 310–327, 2012.

[6] J. H. Jafarian, E. Al-Shaer, and Q. Duan, “Spatio-temporal address mutation for proactive cyber agility against sophisticated attackers,” *Proceedings of MTD workshop at CCS’14*, pp. 69–78, 2014.

[7] N. Hoque, M. H. Bhuyan, R. Baishya, D. Bhattacharyya, and J. Kalita, “Network attacks: Taxonomy, tools and systems,” *Journal of Network and Computer Applications*, vol. 40, pp. 307–324, 2014.

[8] Y.-B. Luo, B.-S. Wang, X.-F. Wang, X.- F. Hu, G.-L. Cai and H. Sun, “RPAH: Random Port and Address Hopping for Thwarting Internal and External Adversaries”, *Proceedings of the 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications,* pp. 263–270, 2015.

[9] A. R. Chavez, W. M. S. Stout and S. Peisert, “Techniques for the Dynamic Randomization of Network Attributes”, *Security Technology (ICCST), 2015 International Carnahan Conference,* Taipei, pp. 1-6, 2015.

[10] Y. Li, R. Dai and J. Zhang, "Morphing communications of Cyber-Physical Systems towards moving-target defense," *2014 IEEE International Conference on Communications (ICC),* Sydney, NSW, pp. 592-598, 2014.

[11] A. Ashok, Pengyuan Wang, M. Brown and M. Govindarasu, "Experimental evaluation of cyber-attacks on Automatic Generation Control using a CPS Security Testbed," *2015 IEEE Power & Energy Society General Meeting*, Denver, CO, pp. 1-5, 2015.

[12] S. Sridhar and M. Govindarasu, "Data integrity attacks and their impacts on SCADA control system", *in IEEE Power and Energy Society General Meeting*, pp. 1-6, 2010.

[13] A. Hahn, A. Ashok, S. Sridhar and M. Govindarasu, "Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid", *Smart Grid, IEEE Transactions on*, vol. 4, no. 2, pp. 847-855, 2013.

[14] Govindarasu, M. and Benzel, T. and Hahn, A., "Smart Energy CPS - CPS Security Testbed Federation for Coordinated Cyber Attack/Defense Experimentation," June 2014. URL: http://smartamerica.org/news/iowastate-researchers-to-demonstrate-cyber-physical-security-testbed-forpower-grid-at-smartamerica-challenge-expo/.

[15] B. Van Leeuwen, W. M. S. Stout and V. Urias, "Operational cost of deploying Moving Target Defenses defensive work factors," *Military Communications Conference, MILCOM 2015 - IEEE*, Tampa, FL, pp. 966-971, 2015.

[16] S. Groat, M. Dunlop, W. Urbanksi, R. Marchany and J. Tront, "Using an IPv6 moving target defense to protect the Smart Grid," *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, Washington, DC, pp. 1-7, 2012.

[17] R. Langner, "Stuxnet: Dissecting a Cyberwarfare Weapon," in *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49-51, May-June 2011.

[18] A. Hahn, A. Ashok, S. Sridhar and M. Govindarasu, "Cyber-Physical Security Testbeds: Architecture, Application, and Evaluation for Smart Grid," in *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 847-855, June 2013.

[19] *Upgrade Coming to Grid Cybersecurity in U.S.*
(2016, April 20). Retrieved from http://spectrum.ieee.org/energy/the-smarter-grid/upgrade-coming-to-grid-cybersecurity-in-us

[20] D. Xinshu, L. Hui, T. Rui, K. Ravishankar, K. Zbigniew, "*Software-Defined Networking for Smart Grid Resilience: Opportunities and Challenges*", CPSS'15, pg: 61-68 , April 14-17, 2015, Singapore doi:10.1145/2732198.2732203
URL: http://dx.doi.org/10.1145/2732198.2732203

[21] P. Kampanakis, H. Perros, T. Beyene, "SDN-based solutions for Moving Target Defense network protection," *A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on* , vol., no., pp.1,6, 19-19 June 2014 doi: 10.1109/WoWMoM.2014.6918979

URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6918979&isnumber=6918912

[22] "*OpenFlow Random Host Mutation: Transparent Moving Target Defense using Software Defined Networking",* HotSDN '12 Proceedings of the first workshop on Hot topics in software defined networks, ACM New York, USA doi:10.1145/2342441.2342467

[23] M. Rahman, E. Al-Shaer and R. B. Bobba, *"Moving Target Defense for Hardening the Security of the Power System State Estimation"*, ACM, 2014

[24] M. Carvalho and R. Ford, *"Moving Target Defenses for Computer Networks",* Page no. 73-76, IEEE Security & Privacy, Mar 2014

[25] P. Kampanakis, H. Perros, T. Beyene, *"SDN-based solutions for Moving Target Defense network protection",* WOWMOM, 2014, 2014 IEEE 15th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 2014 IEEE 15th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM) 2014, pp. 1-6, doi:10.1109/WoWMoM.2014.6918979

[26] H. Okhravi, M.A. Rabe, T.J. Mayberry, W.G. Leonard, T.R. Hobson, D. Bigelow, W.W. Streilein, *"Survey of Cyber Moving Targets"*, Technical Report, MIT Lincoln Laboratory, 2013.

[27] H. Shacham and M. Page and B. Pfaff and E. Goh and N. Modadugu and D. Boneh,*"On the Effectiveness of Address-Space Randomization"*, CCS 2004.

[28] G. Duarte, *"Anatomy of a Program in Memory",*
http://duartes.org/gustavo/blog/post/anatomy-of-a-program-in-memory/
www.cs.utexas.edu/~shmat/courses/cs380s_fall09/04aslr.ppt

[29] I. Unruh, A. G. Bardas, R. Zhuang, X. Ou, and S. A. DeLoach, *"Compiling abstract specifications into concrete systems: bringing order to the cloud",* In Proceedings of the 28th USENIX conference on Large Installation System Administration (LISA'14). USENIX Association, Berkeley, CA, USA, 17-33.

[30] R. Zhuang, A. G. Bardas, S. A. DeLoach and X. Ou, *"A Theory of Cyber Attacks -- A Step Towards Analyzing MTD Systems"*, in CCS 2015 MTD Workshop, Denver, CO, US, October, 2015.

[31] R. Zhuang, S. A. DeLoach and X. Ou , "*Towards a theory of moving target defense",* in First ACM Workshop on Moving Target Defense (MTD 2014), Scottsdale, Arizona, USA, November, 2014.

[32] R. Zhuang, S. A. DeLoach and X. Ou,*"A model for analyzing the effect of moving target defenses on enterprise networks",* 9th Cyber and Information Security Research Conference (CSIRC), Oak Ridge, Tennessee, USA, April, 2014

[33] S. DeLoach, X. Ou, R. Zhuang, S. Zhang, I. U. Abmann, N. Bencomo, G. Blair, B. H. C. Cheng, R. France, *"Model-driven, moving-target defense for enterprise network security"*, State-of-the-Art Survey Volume on Models @run.time. Springer LNCS, Volume 8378, 2014, pp 137-161.

[34] R. Zhuang, S. Zhang, A. G. Bardas, S. A. DeLoach, X. Ou, and A. Singhal, *"Investigating the application of moving target defenses to network security",* 6th International Symposium on Resilient Control Systems (ISRCS), San Francisco, CA, August, 2013.

[35] J. Yackoski, J. Li, S. A. DeLoach, and X. Ou, "*Mission-oriented moving target defense based on cryptographically strong network dynamics",* in the 8th Annual Cyber Security and Information Intelligence Research Workshop (CSIIRW), Oak Ridge, TN, Jan 2013.

[36] R. Zhuang, S. Zhang, S. A. DeLoach, X. Ou, and A. Singhal, *"Simulation-based approaches to studying effectiveness of moving-target network defense",* in the National Symposium on Moving Target Research, Annapolis, MD, USA, June, 2012.

[37] Moving Target Defense,
Retrieved from https://www.dhs.gov/science-and-technology/csd-mtd