Towards Moving Target Defense (MTD)-Enabled Communication Protocols for Smart Grids: A DNP3 Use Case

Raveendra Babu Ponnuru
Dept. of CIS
Virginia Military Institute
Lexington, VA, USA
raveendrap@vt.edu

Rajarshi Mukherjee
Dept. of ECE
Virginia Tech
Arlington, USA
rajarshim13@vt.edu

Mohammed Azab
Dept. of CIS
Virginia Military Institute
Lexington, VA, USA
mazab@vt.edu

Thidapat Chantem
Dept. of ECE
Virginia Tech
Blacksburg, VA, USA
tchantem@vt.edu

***Abstract*—Smart grids rely on real-time communication protocols to monitor and control distributed assets, with the Distributed Network Protocol version 3 (DNP3) being one of the most widely adopted. However, DNP3 remains vulnerable to timing manipulation, replay, and denial-of-service attacks due to its deterministic polling cycles and quality-of-service (QoS) requirements. Conventional defenses, such as encryption and firewalls, offer limited protection against these timing-sensitive threats. To address this gap, we propose a Moving Target Defense (MTD) framework that dynamically reconfigures DNP3 communication while preserving its QoS (percentage of deadlines met). The framework integrates defense management, protocol-level manipulation, software-defined networking, and TCP-level strategies such as port hopping and IP shuffling. Experimental validation on a Raspberry Pi-based testbed demonstrates that MTD enhances resilience and reduces attack effectiveness while maintaining QoS, thereby establishing MTD as a practical and adaptive defense for smart grid communications.**

***Index Terms*—Moving target defense, DNP3, SCADA, Real-time systems, Security Attacks**

I. Introduction

Modern electric power infrastructures are rapidly evolving into smart grids, characterized by the integration of distributed energy resources, advanced monitoring systems, and automated control mechanisms. A key component of this transformation is the dependence on real-time communication protocols. These protocols enable supervisory control and data acquisition (SCADA) systems to coordinate the exchange of telemetry and control signals among geographically distributed assets [1]. Among these protocols, the DNP3 has become one of the most widely used standards, ensuring reliable interoperability between control centers, remote terminal units (RTUs), intelligent electronic devices (IEDs), and field equipment [2]. DNP3 is crucial for executing operations such as telemetry reporting, breaker control, and fault isolation, all of which must be completed in a timely manner [3]. In parallel, the IEC 61850 standard has gained traction in modern deployments by enabling object-oriented data modeling and high-speed substation automation. Unlike DNP3, which focuses on reliable data acquisition and control, IEC 61850 emphasizes semantic interoperability and supports advanced services such as GOOSE and sampled values, making it attractive for utilities modernizing their infrastructures [4].

Smart grid applications require deterministic execution, in contrast to general-purpose IT systems, where occasional delays may only result in degraded performance. A millisecond delay is necessary between protective relays, telemetry updates, and acknowledgments to synchronize master-outstation communication in near-real time [5]. However, this need for precision also makes smart grid protocols vulnerable to cyberattacks. Due to their predictability, deterministic polling cycles, and QoS requirements, DNP3 and similar protocols have become increasingly attractive targets for attackers, leading to operational disruptions, equipment damage, and cascading outages [6]. While firewalls, intrusion detection systems, and cryptographic protections are critical, they are often insufficient against adversaries who exploit timing-sensitive behaviors [7].

This vulnerability presents a fundamental challenge: securing real-time protocols without compromising their QoS. To address this issue, this work investigates the potential of MTD as an adaptive strategy. MTD involves introducing controlled, dynamic changes to the communication surface—such as altering port numbers, routes, or protocol parameters—thereby increasing uncertainty for attackers while maintaining operational continuity [8].

The contributions of this work are fourfold:

1) We analyze the QoS and operational requirements of DNP3 as a representative smart grid protocol, highlighting how these properties both enable reliable control and expose vulnerabilities.
2) We survey and categorize potential cyberattacks on real-time protocols, explicitly mapping them to the temporal properties of DNP3 traffic.
3) We propose a conceptual framework that integrates MTD strategies into real-time communication systems. This includes a mapping between specific attack vectors and MTD techniques, with an emphasis on benefits and trade-offs.
4) We implement a part of the proposed framework on a Raspberry Pi testbed and conduct experiments. Results indicate that MTD mechanisms effectively reduce

This paper received support from the Commonwealth Cyber Initiative, an investment in the advancement of cyber R&D, innovation, and workforce development. For more information about CCI, visit: www.cyberinitiative.org

adversarial success rates while meeting QoS with only moderate performance overhead.

II. Background

A. Smart Grids as Real-Time Systems

Real-time systems (RTS) are characterized by the enforcement of both temporal correctness and logical correctness. In such systems, the timeliness of task execution is integral. Real-time systems are typically classified as: *hard real-time systems*, which require absolute adherence to deadlines [9]; *soft real-time systems*, which tolerate a prespecified number of deadline violations [10]; and *firm real-time systems*, where occasional deadline misses are tolerated but late results are discarded. In mission-critical infrastructures such as smart grids, even a single missed deadline can cause large-scale failures, emphasizing the need for communication protocols that can maintain required QoS levels.

B. DNP3 Protocol

DNP3 is widely deployed in SCADA systems and smart grid infrastructures to enable communication between master stations and outstations. Its layered architecture—comprising application, transport, and data link layers—supports both serial and TCP/IP communication, thereby ensuring interoperability across heterogeneous devices. As illustrated in Fig. 1, each DNP3 frame contains a header with synchronization, length, control, addressing, and CRC fields, followed by a data session consisting of blocks (1–16 bytes each) with individual CRCs for fine-grained error detection. These design choices enhance robustness in noisy industrial environments and support advanced features such as time synchronization, time-stamped event reporting, broadcast messaging, and unsolicited responses from RTUs, which reduce polling overhead and improve efficiency. At the same time, DNP3 underpins numerous timing-critical operations whose behaviors must remained deterministic even while under attack: telemetry must be delivered within pre-specified intervals, breaker commands have deadlines for fault isolation, and protective relays must respond to trip signals within milliseconds. Even minor delays in acknowledgments or event messages can disrupt synchronization between master and outstation systems. Although DNP3 Secure Authentication (DNP3-SA) improves confidentiality and integrity, many legacy deployments persist without these extensions, leaving the protocol vulnerable to spoofing, modification, replay, and eavesdropping. Consequently, while DNP3 provides reliable and interoperable communication, its predictability and QoS requirements continue to present security risks for mission-critical infrastructure.

III. Threat Landscape for Real-Time Protocols

Having established the deterministic nature of DNP3, we now examine threats that specifically exploit these properties. The most significant attack classes are timing attacks, replay and injection, man-in-the-middle (MITM), and denial-of-service (DoS). While such attacks may only cause degraded performance in conventional IT systems, their impact in smart

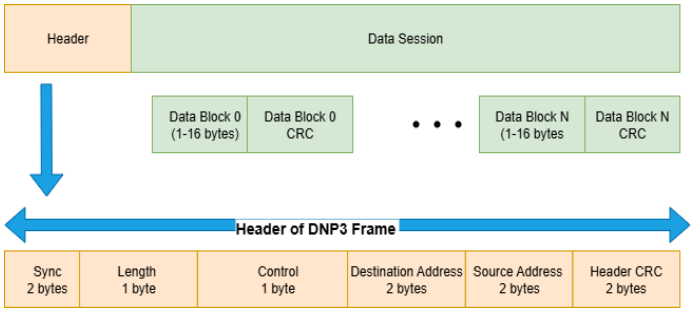

Fig. 1. Structure of DNP3 frame

grids can include operational instability, equipment damage, and cascading blackouts.

A. Timing Attacks

DNP3 traffic is highly latency-sensitive. Operations such as fault detection and relay tripping must occur within milliseconds. If an adversary injects delays or jitter into trip messages, relays may miss protection windows, allowing faults to propagate. Similarly, delayed or reordered status updates may desynchronize the MTU's view of the grid state, undermining automated logic. Even millisecond-level deviations that are tolerable in IT systems can destabilize power grid operations [11].

B. Replay and Injection Attacks

The deterministic polling cycles and repetitive message formats in DNP3 make it susceptible to replay and injection. A replayed "open breaker" command can unexpectedly disconnect loads, while injected false telemetry may mask faults and mislead operators. Without strong freshness guarantees, such attacks may go undetected, corrupting both control flows and situational awareness [11].

C. Man-in-the-Middle (MITM) Attacks

MITM attackers can alter or suppress DNP3 traffic in real time. Manipulated measurements, such as voltage or frequency, may conceal instability from the control center, while modified breaker commands can trigger unauthorized switching. Even subtle changes to acknowledgments may desynchronize master–outstation communication, leading to unsafe conditions or cascading failures [12].

D. Denial-of-Service (DoS) Attacks

Resource-limited DNP3 devices are highly susceptible to DoS. Flooding links with spurious or malformed DNP3 requests can block critical traffic. For example, if a trip command is delayed by congestion, a faulted line may remain energized, risking equipment damage. Even modest DoS campaigns can cause missed actions and endanger system safety [13]. Taken together, these threats highlight the inadequacy of static defenses and motivate adaptive techniques such as MTD. Table I summarizes how key attack types exploit inherent properties of DNP3 and their resulting consequences.

TABLE I
MAPPING OF ATTACK TYPES TO DNP3 VULNERABILITIES AND CONSEQUENCES IN GRID OPERATIONS

Attack Type	Exploited DNP3 Property	Example Scenario	Consequences
Timing Attacks	Deterministic scheduling and millisecond-scale deadlines	Injecting jitter into a "trip" command or reordering telemetry frames	Relay misses actuation deadline; MTU processes stale state, leading to incorrect automated control decisions
Replay Attacks	Predictable polling cycles; absence of freshness and anti-replay checks	Replaying a valid "open breaker" command or injecting falsified "healthy feeder" telemetry	Unauthorized breaker actuation interrupts service; falsified telemetry conceals faults, delaying isolation and escalating failures
MITM Attacks	Unencrypted, mission-critical master–outstation exchanges	Intercepting and altering voltage/frequency reports or modifying breaker control commands	Control center misinformed about grid state; manipulated switching destabilizes operations; altered acknowledgments desynchronize communication
DoS Attacks	Bandwidth- and CPU-limited RTUs/IEDs with QoS requirements	Flooding channels with spurious or malformed requests to exhaust resources	Critical trip commands delayed or dropped; telemetry suppressed, leaving MTU blind to real-time conditions and preventing timely fault remediation

IV. THREAT MODEL

We assume an adversary capable of passive monitoring, active injection, traffic manipulation (MITM), and resource exhaustion. The attacker may compromise intermediate nodes or exploit insecure links between SCADA components. However, physical access to field devices is out of scope. The adversary's primary goal is to exploit timing predictability to disrupt operations, degrade availability, or cause unsafe switching actions. Defenses must therefore provide unpredictability without compromising QoS.

V. MOVING TARGET DEFENSE AS A SOLUTION

MTD shifts the attack surface of a system by introducing controlled dynamism into network and protocol operations, thereby increasing uncertainty and complexity for adversaries. Unlike static defenses that rely on fixed configurations, MTD employs strategies such as address randomization and dynamic protocol transformations to reduce the effectiveness of reconnaissance and persistent attacks. However, implementing MTD in smart grid protocols such as DNP3 presents unique challenges. Although these strategies improve resilience, they must not compromise QoS requirements. The effectiveness of MTD in this context therefore depends on carefully balancing unpredictability for adversaries with determinism for operators: excessive dynamism risks violating too many deadlines, while insufficient adaptation leaves communication patterns predictable. Thus, careful calibration is essential to ensure that security improvements remain compatible with the strict temporal requirements of the mission-critical infrastructure.

To address these challenges, we propose an MTD-enabled framework that dynamically reconfigures DNP3 communication while preserving QoS. The framework, shown in Fig. 2, consists of five functional modules: the Defense Management Unit (DMU), DNP3 manipulation, SCADA components, SDN infrastructure, and TCP manipulation. Each contributes to adaptive resilience by introducing unpredictability into communication flows while ensuring compliance with QoS constraints.

A. Defense Management Unit

The *Defense Management Unit* serves as the central decision-making and orchestration layer. It translates situational awareness into adaptive defense actions.

MTD Strategy Synthesis consists of a few builders: the *Shuffling Strategy Builder*, which produces IP, port, and path randomizations; the *Diversity Strategy Builder*, which introduces heterogeneity into message formats and communication patterns; and the *Deception Strategy Builder*, which designs honeypot and decoy strategies. This sub-module depends on inputs from any Intrusion Detection System (IDS) to develop and assign appropriate reactionary strategies to counter specific threats. While each system will possess a standard proactive set of MTD strategies, this sub-module will help to add reactionary strategies on top of standard configurations.

Rule Enforcement and Execution is responsible for taking the high-level security policies suggested by the synthesis module converting them to low-level enforceable rules, and then carrying them out across the network. The *Flow Rule Enforcer* dynamically updates the necessary tables in SDN switches and routers. The *NAT/Proxy Configurator* manages address and port bindings to ensure that sessions don't abruptly terminate, and the *Honeypot Orchestrator* deploys and manages decoy nodes with an intention to lure potential adversarial traffic to these nodes.

B. DNP3 Manipulation Module

The *DNP3 Manipulation Module* integrates deception, decoy-based defenses, and protocol-level variability to increase adversarial uncertainty and strengthen the resilience of SCADA communications. Positioned between the SDN infrastructure and critical SCADA components, this module serves as an active defense layer capable of intercepting, redirecting, and mutating traffic before it reaches mission-critical devices. Future work will extend this module by integrating dynamic honeypot orchestration, cross-layer deception strategies, and automated intelligence feedback loops to further enhance adaptability in mission-critical infrastructures.

Virtual SCADA Components emulate authentic endpoints using shufflers, traffic redirection proxies, and ephemeral con-

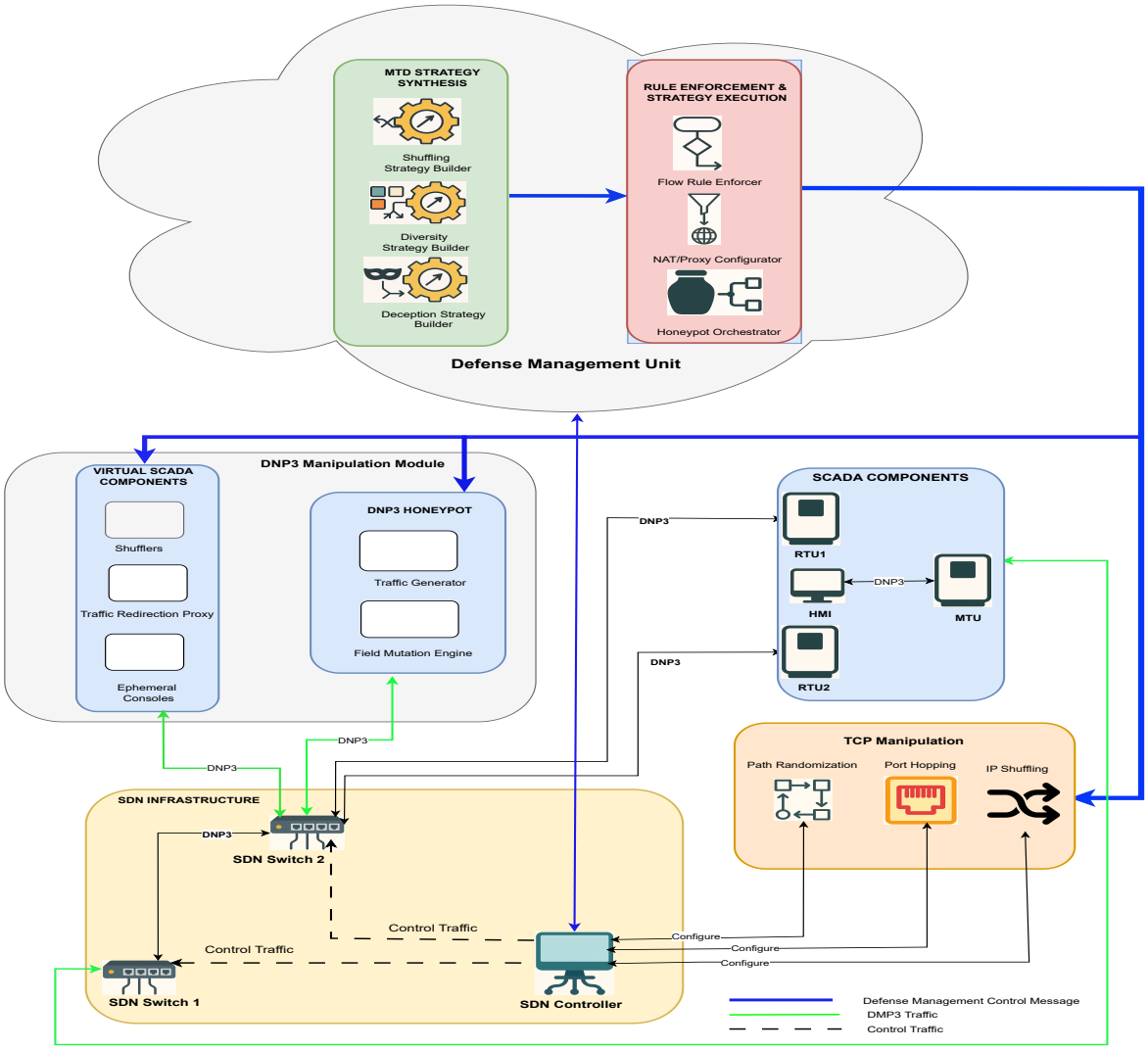

Fig. 2. Proposed MTD-Enabled DNP3 Communication Framework for Smart Grid Real-Time Systems

soles. These elements mislead attackers, increase reconnaissance costs, and divert adversarial actions away from mission-critical devices.

DNP3 Honeypot incorporates a *Traffic Generator* and a *Field Mutation Engine*, enabling realistic simulation of DNP3 exchanges. By mutating message fields and introducing controlled variability, the honeypot both resists replay attacks and records malicious interactions for forensic analysis.

C. SCADA Components Module

The *SCADA Components* module represents *RTUs*, *IEDs*, *MTUs*, and *HMIs*. Within the framework, their communications remain shielded by surrounding MTD strategies, ensuring that legitimate telemetry and control functions are preserved even during attacks. This ensures that essential operations, such as relay tripping and fault detection, continue to meet their QoS requirements.

D. SDN Infrastructure Modules

The software-defined networking (SDN) infrastructure serves as the programmable backbone of the MTD framework, allowing rapid enforcement of dynamic security strategies. By separating the control plane from the data plane, SDN offers global network visibility while adapting forwarding rules on demand. This separation is crucial in smart grid environments, where defense mechanisms must be implemented without affecting the determinism of DNP3.

SDN Controller: The SDN Controller acts as the central authority responsible for orchestrating configuration changes across the communication infrastructure. It receives policy directives and MTD strategies synthesized in the control plane,

translates them into actionable flow rules, and disseminates these rules to SDN switches. Crucially, the controller continuously monitors performance metrics such as latency and jitter, ensuring that any reconfiguration does not affect the ability of DNP3 to provide deterministic service when it comes to protective relays, breaker commands, and telemetry updates. The controller thus serves as the bridge between high-level defense logic and real-time enforcement, maintaining a delicate balance between unpredictability for adversaries and determinism for legitimate operations.

SDN Switches: SDN switches function as the distributed enforcement points where the controller's instructions are realized. Each switch maintains a flow table that can be dynamically updated to reflect MTD strategies such as altered paths, reassigned ports, or reallocated addresses. These updates occur at line speed, minimizing additional latency and ensuring that DNP3 traffic continues to meet its QoS requirements. By enabling per-flow granularity, SDN switches also allow differentiated treatment of traffic classes–for example, ensuring that trip commands receive a higher priority while telemetry or status messages are reconfigured with degraded QoS level. In this way, the switches operationalize MTD in a manner that is both flexible and QoS aware.

E. TCP Manipulation Module

The *TCP Manipulation Module* embodies concrete MTD strategies that are observable to adversaries:

Path Randomization: Path randomization periodically alters the communication routes taken by DNP3 traffic between master stations and outstations. By dynamically reassigning flows across alternative paths on the transport layer, the framework complicates adversarial reconnaissance and prevents persistent man-in-the-middle positioning. By having such reorganization events occur at randomized intervals, attackers are further prevented from being able to anticipate future route changes. Since routing changes are selected with awareness of QoS requirements, critical operations such as relay tripping or breaker actuation remain unaffected.

Port Hopping: Port hopping introduces temporal variability in the transport-layer ports used by DNP3 sessions. This disrupts adversaries who rely on fixed port numbers to inject commands, replay messages, or flood endpoints. By continuously reassigning ports at controlled intervals, the mechanism forces attackers to repeatedly re-identify valid communication channels, thereby increasing their workload and reducing the likelihood of successful exploitation. At the same time, synchronization between master and outstation devices is preserved through SDN-coordinated updates, preventing disruptions to legitimate sessions.

IP Shuffling: IP address shuffling periodically reassigns the addresses of DNP3 devices, invalidating adversarial reconnaissance data and hindering targeted attacks. Attackers that rely on static mappings of RTUs, IEDs, or master stations are forced into repeated discovery cycles, increasing their exposure to detection. Address reassignment is performed within predefined pools under SDN controller supervision to prevent desynchronization of legitimate communications. This controller is responsible for sneding synchronized updates, revised flow tables, and and ARP announcements

VI. Case Study: DNP3 with MTD

The following experiments simulate the communication between a master station and its outstation in a typical smart grid which uses DNP3 for information exchange. The goal of these trials is to show how an MTD enabled DNP3 based network might be able to rapidly adopt different strategies based on inputs from the previously discussed modules, and deal with different attack vectors. The communication channel is subjected to various attacks as mentioned above, and the efficacy of MTD techniques is evaluated against such attacks.

A. Experiment Design

The outstation code is hosted on a Raspberry Pi 4 (RPi) that reports the feeder power load (in kW) and bus voltage (in kV) to the master station alongside the present condition of the circuit breaker. The master station is hosted on a Windows PC, and its role is to periodically poll the outstations for inputs, and take control decisions based on these inputs by employing a deterministic logic. Additionally, this station maintains a log of all communications, as well as the response times to every request, and potential timing violations. For ease of implementation, the stations communicate over TCP using a lightweight protocol to imitate DNP3 based communication by using different classes of polling (fast and slow polls), commands to control actuators, and such. Each poll request has a deadline. For the ease of evaluation, this work assumes that an IDS exists that detects any attack on this network and intimates the strategy module so that it can recommend the appropriate strategy to best counter the threat.

B. Attack scenarios

For the purposes of this experiment, three different attack scenarios are simulated which aim to attack critical vulnerabilities in a given network's defenses. More details about these scenarios can be found listed below.

1) DoS Attacks: After establishing the aforementioned master-outstation communication framework, a DoS attack is carried out to flood the data port with synthetic payloads and connection requests. A synthetic workload generator launches a combination of flood and Slowloris [14] style connection requests towards the port that is being used by the network. The goal of this exercise is to prevent poll requests from being serviced within their deadlines, thus resulting in performance degradation of the simulated network. In order to mitigate this, a port shuffle is requested with the master station issuing a rotate command to the outstation over a dedicated control channel. This allows the nodes to communicate on the secondary port till the next reshuffle is requested.

2) Replay Attack: A replay attack is simulated by first collecting valid responses during the polling phase, before modifying its identifier fields to trick the master station into believing it to be live data from the outstation. Such attacks

exploit systems which do not enforce checks to validate if a syntactically accurate data packet is actually stale. The actual attack is designed to stochastically inject a replayed frame with an attacker defined replay rate of around 35%. This ensured that the master station received a controlled mix of real and adversarial data with some built in randomness. As a way to defend against such attacks, HMAC verification as proposed in [15] is carried out, thereby binding each message reply to a unique cryptographic tag(derived from the message sequence number and the particular device's MAC address). Since this tag continuously changes, the attack surface can be imagined to be continuously moving as well. This helps protect against stale data being passed through the system.

3) Code Injection Attack: In order to evaluate the dangers posed by a code injection attack, we attempted to inject intentionally lowered bus voltage values with the aim to generate improper circuit breaker commands. As a way to defend against such attacks, this work adopted a diversity based approach that utilized double channel decoding to check for improper tampering by building on the idea presented in [16]. The voltage value at each reading is encoded into two independent channels by using time varying parameters that are generated by a shared random seed, or secret. The master station then reconstructs the value using this shared secret. An attacker must thus take control over both channels to defeat this setup, since a disagreement between channels leads to rejection of results.

VII. Results and Discussion

Table II demonstrates the benefits of incorporating an MTD based shuffling approach to counter DoS attacks. When a determined attacker targeted the port used for communication, the baseline version had 74% of its deterministic tasks miss their deadlines. When a port shuffling algorithm is used to keep rotating the ports used for communication, the overall deadline misses were reduced to around 55% of tasks. This represents a relative improvement of around 25% with shuffling engaged. While some of these deadline misses are due to a persistent attacker who aims to reacquire and attack the port under use, the actual act of shuffling imposes an overhead as well that impacts the ability to service certain requests. This is evident from the drop in the overall throughput and the increase in the average polling time.

TABLE II
Comparison of Baseline and MTD-Equipped System Under DoS Attack

Metric	Baseline	MTD	Observation
Deadlines missed (%)	74	55	Fewer deadlines missed
Throughput (msg/s)	1.71	1.15	Overhead reduces throughput
Avg. polling time (ms)	82.4	109.2	Higher service time with MTD

In Fig. 3, the results of a replay attack are considered. Both tests are carried out across a fixed time period and rate of replayed frames. Since the baseline configuration has no direct defense against such a threat, it accepts all the replayed packets as legitimate. The bottom image shows the effectiveness of the verification scheme in rejecting the replayed sequences. The added security measures increase the average response time of polls from `35.5 ms` to `44.5 ms`. In around `2.7%` of instances, a failed replay attack, and the additional verification overhead does lead to some valid data frames missing their deadlines.

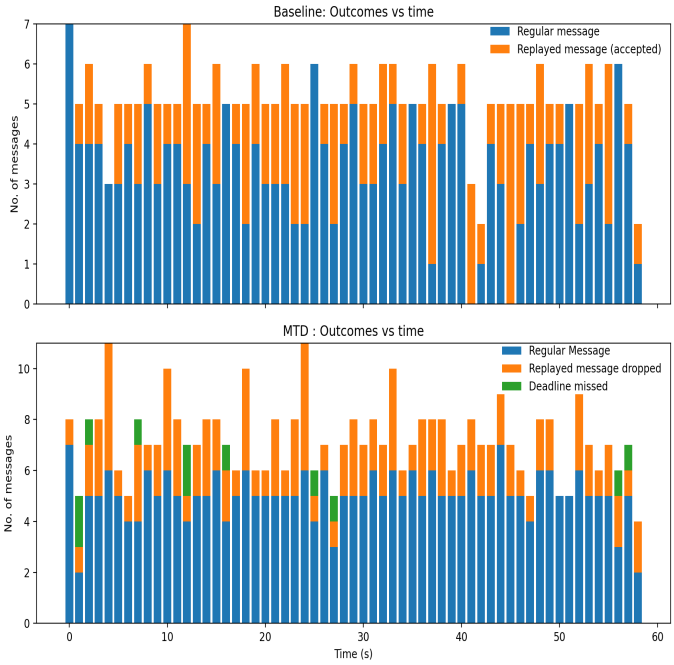

Fig. 3. Comparing replay attacks under baseline and MTD based configurations. The top graph is indicative of a basic system that does not check for replayed messages. The bottom graph shows the regular messages, the replay messages that are caught and dropped, and the occasional deadline overruns that occur as a result of enhanced security.

Fig. 4 represents the third batch of experiments to showcase code injection attacks, and how diversity based approaches can help stop them. The plot shows the bus voltages across the duration of the experiment, with the actual voltage reading as reported by the outstation in blue, the readings accepted by the baseline model in red, and the voltage recorded when MTD is enabled in green. The dotted line represents the trigger voltage for the simulated circuit breaker. While the baseline model traces the actual voltage well for the most part, strategically placed data injections drive the system to engage the breakers when the red line dips below the triggering threshold. The simple dual channel decoding technique (only samples where both channels agree with each other are accepted) used by the MTD enabled system does show a slight difference from the actual bus voltages, but is importantly able to catch all attempts to tamper with the voltage levels.

These results highlight the need to talk about the security-QoS tradeoff. MTD increases both the overhead on the host system and the resources that an attacker must expend to launch a successful attack. For system designers, this balancing act of being able to provide adequate security while still being able to deliver the highest QoS is particularly

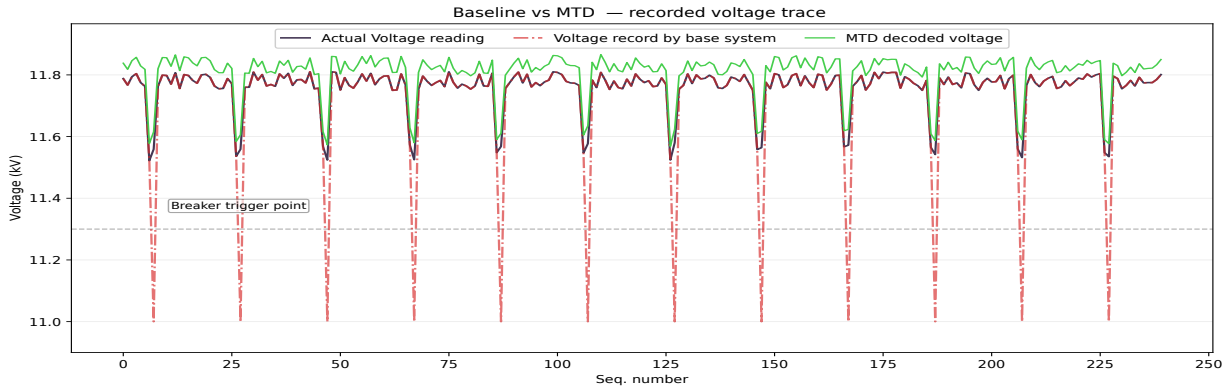

Fig. 4. Comparing code injection attacks under baseline and MTD based configurations. An attacker attempts to force erroneous activation of the circuit breaker based on injected voltage levels. The base system accepts the injected data and triggers the circuit breaker at the attacker's whims. The MTD enabled approach does carry some overhead, but by ensuring diversity of communication channels, it avoids false breaker triggers.

challenging. The role that studies like this play is to provide a measure of exactly how much the aforementioned overhead might be, how much of a deterrent might these defenses be for persistent attacker, and how is the increase in security worth the drop in performance.

VIII. Conclusion and Future Work

This work confronts significant security challenges in real-time communication within smart grids, particularly by examining vulnerabilities in DNP3 against various cyberattacks. Recognizing the imperative of resilience, we introduced a MTD communication framework designed to adaptively reconfigure DNP3 while safeguarding real-time performance. This innovative framework incorporates defense management, DNP3 reengineering, and dynamic strategies such as port hopping and IP shuffling to enhance security. Our experimental results on a RPi testbed demonstrate that MTD drastically lowers the success rates of different attack vectors. While the mechanisms introduce a reasonable performance overhead, the benefits of increased resilience outweigh the costs, making this approach invaluable for mission-critical infrastructure within smart grids. With an eye towards the future, more work is recommended in building out the entire framework on a working test-bed that mirrors actual smart grids more closely. This would allow for tests under more realistic workloads, with extensive attack scenarios, and dynamic MTD strategies generated by Large Language Model based agents. Lastly, focus needs to be directed towards quantifying the security-QoS tradeoff to allow for a holistic review of potential security methods.

References

[1] L.-H. Nguyen, V.-L. Nguyen, R.-H. Hwang, J.-J. Kuo, Y.-W. Chen, C.-C. Huang, and P.-I. Pan, "Towards secured smart grid 2.0: exploring security threats, protection models, and challenges," *IEEE Communications Surveys & Tutorials*, 2024.

[2] IEEE Power and Energy Society, *IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3)*, Std. 1815-2012, 2012.

[3] Q. S. Qassim, M. A. Ali, and N. M. Tahir, "Security analysis of dnp3 protocol in scada system," in *2023 IEEE 13th International Conference on Control System, Computing and Engineering (ICCSCE)*. IEEE, 2023, pp. 314–319.

[4] R. E. Mackiewicz, "Overview of iec 61850 and benefits," in *2006 IEEE Power Engineering Society General Meeting*. IEEE, 2006, pp. 8–pp.

[5] D. Gutierrez-Rojas, P. H. J. Nardelli, G. Mendes, and P. Popovski, "Review of the state of the art on adaptive protection for microgrids based on communications," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 1539–1552, 2020.

[6] J. Chen, J. Yan, A. Kemmeugne, M. Kassouf, and M. Debbabi, "Cybersecurity of distributed energy resource systems in the smart grid: A survey," *Applied Energy*, vol. 383, p. 125364, 2025.

[7] J. E. Efiong, B. O. Akinyemi, E. A. Olajubu, G. A. Aderounmu, and J. Degila, "Cyberscada network security analysis model for intrusion detection systems in the smart grid," in *The International Symposium on Computer Science, Digital Economy and Intelligent Systems*. Springer, 2022, pp. 481–499.

[8] S. Sengupta, A. Chowdhary, A. Sabur, A. Alshamrani, D. Huang, and S. Kambhampati, "A survey of moving target defenses for network security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1909–1941, 2020.

[9] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *Journal of the ACM (JACM)*, vol. 20, no. 1, pp. 46–61, 1973.

[10] J. P. Erickson, J. H. Anderson, and B. C. Ward, "Fair lateness scheduling: Reducing maximum lateness in g-edf-like scheduling," *Real-Time Systems*, vol. 50, no. 1, pp. 5–47, 2014.

[11] M. Cheminod, L. Durante, and A. Valenzano, "Review of security issues in industrial networks," *IEEE transactions on industrial informatics*, vol. 9, no. 1, pp. 277–293, 2012.

[12] R. B. Ponnuru, B. Palaniswamy, M. Azab, P. Palmieri, and U. Roedig, "Protecting dnp3-sab (sav6): A quantum-safe hybrid authentication protocol with moving target defense," *IEEE Transactions on Consumer Electronics*, 2025.

[13] M. Marian, A. Cusman, D. Popescu, and D. Ionică, "A dnp3-based scada architecture supporting electronic signatures," in *2019 20th International Carpathian Control Conference (ICCC)*. IEEE, 2019, pp. 1–6.

[14] S. Sabri, N. Ismail, and A. Hazzim, "Slowloris dos attack based simulation," in *IOP Conference series: materials science and engineering*, vol. 1062, no. 1. IOP Publishing, 2021, p. 012029.

[15] C. R. Taylor, C. A. Shue, and N. R. Paul, "A deployable SCADA authentication technique for modern power grids," in *2014 IEEE International Energy Conference (ENERGYCON)*, 2014, pp. 1287–1294. [Online]. Available: https://web.eecs.utk.edu/ pauln/papers/energycon14.pdf

[16] Z. Zhang, R. Deng, D. K. Y. Yau, P. Cheng, and J. Chen, "Analysis of moving target defense against false data injection attacks on power grid," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2320–2335, 2020.