Vladimír Szomosi, Marek Bobček, Július Šimčák, Róbert Štefko, Zsolt Čonka

Robust Detection and Mitigation of PTP Time Synchronization Attacks on PMU Measurements in Digital Substations: A Redundancy-Based Architecture

Digitalizácia energetiky zvyšuje závislosť systémov WAMS a jednotiek PMU na presnej časovej synchronizácii (PTP), čo vytvára priestor pre nebezpečné útoky na časové signály. Táto štúdia dokazuje, že aj minimálna manipulácia s časom spôsobí chybu merania (TVE) nad kritickú hranicu 1 %. Ako riešenie navrhujeme architektúru s trojitou modulárnou redundanciou (TMR) a inovatívnym hlasovacím algoritmom, ktorý nepretržite verifikuje dáta z troch nezávislých serverov. Simulácie potvrdili, že tento mechanizmus dokáže útok efektívne izolovať a udržať presnosť merania v rámci noriem. Ide o kľúčový prvok pre stabilitu a bezpečnosť moderných digitálnych rozvodní voči sofistikovaným kybernetickým hrozbám.

Kľúčové slová: Kybernetická bezpečnosť, Digitálna rozvodňa, Fázorová meracia jednotka, IEEE C37.118, Časová synchronizácia, Redundancia, Celková vektorová chyba

The digitalization of the energy sector increases the reliance of WAMS and PMU units on Precision Time Protocol (PTP), creating a vulnerability to dangerous time signal attacks. This study demonstrates that even minimal time manipulation causes measurement errors (TVE) to exceed the critical 1% threshold. As a solution, we propose a Triple Modular Redundancy (TMR) architecture with an innovative voting algorithm that continuously verifies data from three independent servers. Simulations confirmed that this mechanism can effectively isolate attacks and maintain measurement accuracy within standards. It is a key element for the stability and security of modern digital substations against sophisticated cyber threats. **(Robust Detection and Mitigation of PTP Time Synchronization Attacks on PMU Measurements in Digital Substations: A Redundancy-Based Architecture)**

Keywords: Cybersecurity, Digital Substation, Phasor Measurement Unit, PMU, IEEE C37.118, Time Synchronization, Redundancy, Total Vector Error

I. INTRODUCTION

The global energy sector is currently undergoing a profound digital transformation, moving away from centralized control towards a complex, interconnected Smart Grid architecture. This shift towards digitalization and automation in power systems has fundamentally increased the importance of cybersecurity in critical infrastructure [1, 2]. While these advancements driven by standards such as IEC 61850 provide unprecedented efficiency, they simultaneously expose the system's operational technology (OT) layer to advanced and sophisticated cyber-physical attacks. Protecting the integrity of real-time operational data is no longer merely a matter of IT security but a fundamental requirement for grid stability and national security. This section critically reviews the existing literature, which directly informs our proposed resilience mechanism. We focus on three interconnected areas: the vulnerabilities inherent in Wide Area Measurement Systems (WAMS), the Achilles' heel represented by the Precision Time Protocol (PTP) in digital substations, and the limitations of the state-of-the-art in resilience mechanisms against synchronized, stealthy threats. The review ultimately establishes the precise research gap that our proposed Triple Modular Redundancy (TMR) architecture is designed to address.

II. CYBERSECURITY CHALLENGES IN WAMS AND PMU INTEGRITY

Wide Area Measurement Systems (WAMS), leveraging Phasor Measurement Units (PMUs) and the IEEE C37.118 standard, are essential for real-time monitoring and control in modern grids [3, 4]. The foundational security concern for PMUs is the integrity of their time-stamped data, as any compromise can lead to system instability [5]. The initial work by Štefko et al. provides a comprehensive overview of cybersecurity threats in the energy sector, highlighting sophisticated malware and the general vulnerability of Industrial Control Systems (ICS) [6].

Specific research has focused on the consequences of corrupted synchrophasor data. Adversarial attacks, such as False Data Injection (FDI), have been studied to show how they can manipulate measurements without immediate detection by traditional Bad Data Detection (BDD) algorithms [7, 8]. Furthermore, studies have shown a critical interplay between data quality issues—such as noise and outliers—and cybersecurity threats, suggesting that a successful cyberattack often manifests as a degradation of data quality [9, 10].

III. VULNERABILITY OF TIME SYNCHRONIZATION PROTOCOLS

The prerequisite for accurate PMU operation is sub-microsecond time synchronization, primarily achieved using the Precision Time Protocol (PTP), formalized as IEEE 1588 [11]. Despite its accuracy, PTP was not designed with comprehensive security in mind, making it highly susceptible to various cyber threats [12].

Verified academic sources identify the most dangerous attacks against PTP networks as those initiated by internal actors or those utilizing a Man-in-the-Middle (MITM) technique [13]. Key threats include:

- Delay Attacks: These attacks stealthily manipulate the PTP delay measurements, gradually shifting the PMU's clock and introducing a critical phase error [14].

- Time Reference Attacks (Spoofing): Attacks targeting the Grandmaster clock's source, such as GPS spoofing, can compromise the time reference for the entire WAMS, resulting in widespread, synchronized errors across the grid [15].
- Denial-of-Service (DoS) Attacks: These attacks prevent the PTP synchronization messages from reaching the PMUs, causing their internal clocks to drift freely, quickly violating the required accuracy [16, 17].

The impact of these attacks is directly quantifiable through the Total Vector Error (TVE). Research confirms that even a modest time error exceeding 27 μs (for a 60 Hz system) or 32 μs (for a 50 Hz system) inherently causes the PMU measurement to exceed the strict 1 % TVE limit required by the IEEE C37.118 standard [18, 19]. The critical significance of this limit, however, extends beyond mere metrological accuracy. Once the measured PMU data crosses the 1 % TVE threshold, it becomes not only inaccurate but also potentially hazardous to grid operation. Such compromised synchrophasor data can lead to severe operational consequences, such as the failure of sophisticated grid protection systems (e.g., during islanding detection or directional protection), the incorrect response of control systems to real-time faults, or even trigger cascading failures that threaten the entire transmission system.

IV. EXISTING COUNTERMEASURES AND RESEARCH GAP

Countermeasures against time synchronization attacks generally focus on three strategies:

- Standardized Cryptography: While standards like IEC 62351-9 provide mechanisms for securing synchrophasor data during transmission, purely cryptographic measures have been shown to be insufficient against sophisticated internal delay attacks that do not alter the cryptographic payload [16].
- Advanced Detection Algorithms: Several studies propose intelligent detection methods, such as Kalman Filters, to simultaneously estimate system states and compensate for synchronization errors. These methods aim to computationally correct inaccuracies but introduce complexity and processing latency [8].
- Architectural Redundancy: The concept of N-modular redundancy is a proven fault-tolerant method for mission-critical systems [7]. In the timing domain, this involves using multiple independent time sources to ensure continuity. However, research indicates that simply deploying redundant components (e.g., multiple grandmasters) is insufficient against coordinated attacks that compromise all paths, underscoring the need for an intelligent voting mechanism [13, 20, 21].

Despite the critical nature of time synchronization, there is a lack of practical research that quantitatively validates a robust, consensus-based Triple Modular Redundancy (TMR) architecture with a dedicated voting algorithm against known PTP attacks while demonstrating compliance with the stringent PMU TVE requirements in a simulated digital substation environment. This paper directly addresses this gap by proposing and experimentally verifying a resilient TMR solution.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

V. PROPOSED TRIPLE MODULAR REDUNDANCY (TMR) ARCHITECTURE AND CONSENSUS ALGORITHM

The limitations identified in existing security mechanisms, particularly their inability to guarantee sub-microsecond time integrity against stealthy delay and spoofing attacks [4, 14, 20], necessitate a shift from purely defensive measures to proactive fault-tolerant architectures. This section details the proposed Triple Modular Redundancy (TMR) framework, which leverages hardware redundancy and an intelligent software-based consensus algorithm to ensure continuous, accurate, and secure time synchronization for mission-critical Phasor Measurement Units (PMUs). The design adheres to the principle of resilience, tolerating the failure or malicious compromise of one of its three independent timing sources.

TMR Architecture for Resilient Time Synchronization:

- Triple Redundant Grandmasters (GM1, GM2, GM3): Three commercially available Precision Time Protocol (PTP) Grandmaster clocks, each equipped with its own independent primary time reference (e.g., dedicated GPS antennas or separate high-quality oscillators). This independence is crucial to prevent common-mode failures, such as a single GPS spoofing signal compromising all sources simultaneously [13, 22, 23].
- Independent Network Paths: Each Grandmaster is connected to the PMU time input through an electrically and logically isolated network segment. This isolation mitigates the risk of a single network-based attack (e.g., DoS or packet delay manipulation) from affecting more than one path.
- Triple Slave Clocks (SC): Three dedicated PTP slave devices (SC_1, SC_2, SC_3), one for each network path, receive the time from their respective Grandmasters. These slaves are co-located and synchronized to feed their time reference output into the central voter.
- Centralized Time Voter and Selection Unit: A dedicated, ruggedized computation unit acts as the core decision-maker. It receives the synchronized time signals (T_1, T_2, T_3) from the three slave clocks and executes the Consensus Algorithm to output the validated, final time signal (T_{final}). This unit is implemented on a real-time platform, utilizing either a powerful Field-Programmable Gate Array (FPGA) for nanosecond-level logic execution or a Real-Time Linux (RT-Linux) kernel running on an industrial single-board computer. This choice ensures deterministic execution and low-latency processing, which is critical for maintaining PMU synchronization accuracy [22, 23, 24].

The output T_{final} is then distributed to all PMUs and other time-critical Intelligent Electronic Devices (IEDs) in the substation, guaranteeing a high level of resilience and mitigating the risk of a single point of failure or attack.

The Consensus (Voting) Algorithm:

The effectiveness of the TMR architecture hinges on the robustness of the Consensus Algorithm executed by the Centralized Time Voter. The algorithm is designed to implement a modified majority voting scheme, specifically tailored for the highly precise requirements of synchrophasor applications. It must be tolerant to one Byzantine fault—meaning one source can be malicious (e.g., outputting a time intentionally delayed or spoofed) while the system continues to output the correct time derived from the two remaining non-compromised sources.

The algorithm proceeds in three phases:

- **Difference Calculation**: At every synchronization epoch k, the Voter measures the time difference (offset) between the three received time signals, $T_1(k)$, $T_2(k)$, $T_3(k)$. The differences Δ are calculated:

$$\Delta_{1,2} = |T_1(k) - T_2(k)| \quad (1)$$

$$\Delta_{1,3} = |T_1(k) - T_3(k)| \quad (2)$$

$$\Delta_{2,3} = |T_2(k) - T_3(k)| \quad (3)$$

- **Median and Threshold Selection:** The three-time signals are ordered, and the median value is identified as the initially preferred time. Simultaneously, the Voter utilizes a predefined Maximum Allowable Discrepancy (MAD) threshold, δ_{MAD}. This threshold is conservatively set to δ_{MAD} = 5 μs, which is five times lower than the 27 μs limit that would violate the 1 % TVE requirement for a 60 Hz system. This significant margin ensures that the detection and isolation of a faulty clock occur pre-emptively, long before the resulting time error could compromise synchrophasor data quality.

- **Fault Detection and Selection Logic**: The Voter compares the calculated time differences (Δ) against δ_{MAD}:

  - **Case I: Full Consensus (All Safe):** If all three differences are below δ_{MAD} ($\Delta_{1,2} < \delta_{MAD}$ AND $\Delta_{1,3} < \delta_{MAD}$ AND $\Delta_{2,3} < \delta_{MAD}$), all sources are considered valid. The final output, T_{final}, is set to the arithmetic **mean** of the three times.

$$T_{final} = \frac{T_1 + T_2 + T_3}{3} \quad (4)$$

  - **Case II: One Fault (Byzantine Tolerance):** If only one difference is large (e.g., $\Delta_{1,2} < \delta_{MAD}$ but $\Delta_{1,3} > \delta_{MAD}$ and $\Delta_{2,3} > \delta_{MAD}$), it indicates that one source is an outlier (the compromised clock). The Voter identifies the two sources that are close (e.g., T_1 and T_2) and sets T_{final} to the arithmetic mean of these two synchronized sources. The outlying source is flagged for reporting.

  - **Case III: Total Failure (System Alert):** If two or more differences exceed δ_{MAD}, meaning no two sources are in consensus (e.g., $\Delta_{1,2} > \delta_{MAD}$ AND $\Delta_{1,3} > \delta_{MAD}$), the system is unable to derive a trusted time reference. The Voter triggers a critical system alert and immediately activates a high-precision, temperature-compensated crystal oscillator (TCXO) or oven-controlled crystal oscillator (OCXO) to enter Holdover Mode. This mechanism provides a stable, self-contained time base, ensuring that the PMUs continue to operate with a time drift rate low enough to prevent a 1 % TVE violation for a critical period (e.g., several hours) until the external synchronization issue is resolved.

By using the median/mean of the two closest sources in Case II, the algorithm effectively provides fault masking, ensuring the PMUs continue to receive highly accurate time signals even under a direct cyberattack on one synchronization path.

Flowcharts for the Architecture and Algorithm:

To ensure maximum clarity and transparency of the implementation logic, the key components of the proposed TMR solution are visualized through two flowcharts. These diagrams serve as a visual demonstration of the complex TMR Architecture (Section III.A) and the decision logic of the Consensus Algorithm (Section III.B).

- TMR Architecture Flowchart: Presents the hardware redundancy and data flow, illustrating the path from three independent time Grandmasters through isolated network segments to the Centralized Voter.
- Consensus Algorithm Flowchart: Details the three-stage decision logic (Case I, II, III) which enables the Voter to determine the system status (Full Consensus, One Fault, Total Failure), mask a single Byzantínec fault, and output a continuous, validated time signal (Tfinal).

CONSENSUS TIME VALIDATION (WITH FAULT MASKING)

```
import math
from typing import Tuple, Union
# MAX_ALLOWABLE_DISCREPANCY_US (δ_MAD): This constant defines the maximum deviation (μs)
# that the algorithm tolerates between two good timing sources.
# If the difference is greater, one of the sources is considered faulty or compromised.
MAX_ALLOWABLE_DISCREPANCY_US = 5.0
def consensus_algorithm(t1: float, t2: float, t3: float, delta_mad: float) -> Tuple[Union[float, str], str]:
    """
    Executes the TMR voting algorithm to determine the final, validated time signal.
    ... (the rest of the docstring remains) ...
    """
    # 1. Difference Calculation
    delta_12 = abs(t1 - t2)
    delta_13 = abs(t1 - t3)
    delta_23 = abs(t2 - t3)
    # consensus_count: This variable counts how many of the three possible pairs of signals are in consensus
    # (i.e., their difference is less than δ_MAD).
    consensus_count = 0
    close_pairs = [ ]
    # Consensus verification: |T_i - T_j| < delta_mad
    if delta_12 < delta_mad:
        consensus_count += 1
        close_pairs.append((t1, t2))
    if delta_13 < delta_mad:
        consensus_count += 1
        close_pairs.append((t1, t3))
    if delta_23 < delta_mad:
        consensus_count += 1
        close_pairs.append((t2, t3))
    # Voting Logic:
    # consensus_count == 3 (Case I):
# All three are good; the output is the mean (highest accuracy).
    if consensus_count == 3:
        t_final = (t1 + t2 + t3) / 3
        status = "Case I: Full Consensus. Final time is the mean of all three sources."
        return t_final, status
```

```
    # consensus_count == 1 (Case II):
    # Exactly one pair is good, meaning the third source (the one not belonging to
the consensus pair) is faulty.
 # The algorithm calculates the mean of the two good sources and masks the fault.
    elif consensus_count == 1:
        t_final = sum(close_pairs[0]) / 2
        # Identification of the compromised source for reporting
        outlier = ""
# Check which time source is NOT in the list of close_pairs
if (t1, t2) not in close_pairs and (t2, t1) not in close_pairs: outlier = "T3"
        elif (t1, t3) not in close_pairs and (t3, t1) not in close_pairs: outlier = "T2"
        else: outlier = "T1"
        status = f"Case II: One Fault Detected. Source {outlier} is the outlier. Final
time is the mean of the two closest sources."
        return t_final, status
# --- Example Usage ---
# 1. Case I: Full Consensus (All times are close)
T1_safe, T2_safe, T3_safe = 1000.1, 1000.2, 1000.3
final_time_1, status_1 = consensus_algorithm(T1_safe, T2_safe, T3_safe,
MAX_ALLOWABLE_DISCREPANCY_US)
# 2. Case II: One Attack (T3 is delayed/spoofed by 20 µs)
T1_ok, T2_ok, T3_attack = 2000.1, 2000.3, 2020.0
final_time_2, status_2 = consensus_algorithm(T1_ok, T2_ok, T3_attack,
MAX_ALLOWABLE_DISCREPANCY_US)
# 3. Case III: Two Attacks (No pair in consensus)
T1_attack_A, T2_attack_B, T3_attack_C = 3000.0, 3010.0, 3020.0
final_time_3, status_3 = consensus_algorithm(T1_attack_A, T2_attack_B,
T3_attack_C, MAX_ALLOWABLE_DISCREPANCY_US)
print("--- Test Case 1: Full Consensus ---")
print(f"Inputs: {T1_safe}, {T2_safe}, {T3_safe} µs")
print(f"Output T_final: {final_time_1:.2f} µs")
```

This visual representation forms the basis for the system's implementation and experimental verification. Furthermore, the algorithm is implemented in detail in the accompanying Python code, which serves as a technical illustration of the mentioned logic.

VI. EXPERIMENTAL SETUP AND ATTACK SCENARIO

To quantitatively validate the resilience of the proposed Triple Modular Redundancy (TMR) architecture and its Consensus Algorithm, a specialized Hardware-in-the-Loop (HIL) laboratory testbed was constructed. This environment simulates a critical digital substation scenario, allowing for the injection of realistic cyberattacks against the Precision Time Protocol (PTP) synchronization layer while monitoring the resulting Total Vector Error (TVE) at the Phasor Measurement Unit (PMU) level. This section outlines the components of the testbed, defines the specific attack scenarios, and details the performance metrics used to evaluate the TMR system's effectiveness.

Experimental Testbed Configuration:

The testbed is designed to isolate the timing system and replicate real-world constraints, adhering to standards like IEC 61850 and IEEE C37.118. The configuration is based on the four-component structure of the TMR system (Section III.A):

TMR Timing System: Three Independent Grandmasters (GM1, GM2, GM3): Three high-precision PTP Grandmaster clocks are utilized, each with an independent time reference (e.g., dedicated GPS receivers or highly stable internal oscillators).

Consensus Voter (Embedded System): A dedicated computational platform (e.g., FPGA or Real-Time Linux box) implements the Consensus Algorithm (Section III.B). It receives three-time inputs and outputs the single, validated Tfinal time signal.

Technical PTP Parameters: The TMR system operates using the default power profile (IEEE C37.238/IEC 61850-9-3) with a PTP Sync message rate of 8 messages/s and a Delay_Req rate of 8 messages/s, resulting in a PTP cycle of 125 ms. These parameters are crucial for replicating typical communication loads in digital substations.

Critical Measurement Device:

- Phasor Measurement Unit (PMU): A commercial PMU compliant with IEEE C37.118 is used as the receiver of Tfinal. Its primary function is to measure and record the TVE caused by any time synchronization errors.

Network and Attack Emulator:

- Network Time Server/Emulator: A high-speed network emulator is placed in the communication paths between the Grandmasters and the Voter. This device allows for the precise injection of network impairments and malicious PTP packets (e.g., Man-in-the-Middle attacks) without affecting the actual timing stability of the Grandmasters themselves.

Reference Measurement:

- High-Resolution Time Analyzer: An external, highly stable Rubidium reference clock and time interval analyzer are used to record the time difference (Time Deviation, TDEV) between the PMU's received Tfinal and a true, uncompromised time reference, ensuring measurement objectivity.

Defined Attack Scenarios:

The TMR system is subjected to three primary categories of synchronized and stealthy cyberattacks, all designed to cause a TVE violation in a non-protected PMU:

Stealthy Delay Attack (Byzantine Fault - Case II Test):

- Mechanism: The network emulator is configured to introduce a small, cumulative, and asymmetrical delay into PTP packets originating from one source (GM3). This delay is slowly ramped up (e.g., 1 µs/s) until the time from GM3 exceeds the δMAD tolerance relative to GM1 and GM2.
- Goal: To test the system's ability to detect and mask a single malicious timing source and continue operation using the two non-compromised sources.

GPS Spoofing / Time Reference Attack (Byzantine Fault - Case II Test):

- Mechanism: One Grandmaster (GM1) is intentionally provided with a spoofed GPS signal, causing its clock to suddenly jump or drift significantly. This simulates a targeted attack on the primary source of time synchronization.
- Goal: To test the TMR system's ability to reject a completely falsified time reference and rely on the consensus of the remaining two sources.

Simultaneous Two-Source Attack (Total Failure - Case III Test):

- Mechanism: Both GM1 and GM2 are simultaneously subjected to different, non-consensus-achieving attacks (e.g., GM1 delayed by +15 μs, GM2 delayed by −15 μs).
- Goal: To test the system's ability to correctly identify a Total Failure (Case III), avoid outputting a corrupted mean time, trigger the critical alarm, and successfully revert to the safe Holdover Mode.

Performance Metrics:

The success of the TMR implementation is measured by its performance under attack relative to the strict requirements of synchrophasor technology:

Total Vector Error (TVE) Compliance:

- The primary metric is ensuring that the PMU's TVE derived from Tfinal remains below the 1 % maximum limit defined by IEEE C37.118 throughout the duration of the attack in the Case II scenario. A successful test is one where the unprotected PMU would fail, but the PMU synchronized by Tfinal remains compliant.

Fault Masking Latency:

- Measures the time interval (in PTP synchronization cycles) between the moment the Consensus Algorithm detects the first $\Delta > \delta MAD$ violation and the moment the output Tfinal stabilizes to the mean of the two good sources. Minimal latency is critical to prevent short-term TVE spikes.

Holdover Integrity:

- In the Case III scenario, measures the drift rate of the PMU's TVE after the system reverts to Holdover mode. The system must confirm that the Holdover integrity is maintained for a defined period (e.g., 60 seconds) without violating the TVE threshold.

VII. CONCLUSION

The cybersecurity resilience of the Smart Grid is directly dependent on the integrity of its key components. This paper addressed a critical research gap concerning the absence of a practically validated, fault-tolerant mechanism capable of securing sub-microsecond time synchronization for Phasor Measurement Units (PMUs) against sophisticated attacks on the Precision Time Protocol (PTP). Our review of the related work confirmed that traditional cryptographic and detection methods are ineffective against coordinated delay and spoofing attacks that threaten the Total Vector Error (TVE) limit in WAMS systems.

We proposed and implemented Triple Modular Redundancy (TMR) architecture, which represents a robust shift from defensive to proactive solutions. Crucially, the primary contribution of this work is not mere fault detection (which existing algorithms can often achieve), but the implementation of seamless Fault Masking. The core of this system is the Consensus Algorithm, utilizing modified majority voting logic to achieve tolerance against a single Byzantine fault. Experimental validation conducted within a Hardware-in-the-Loop (HIL) environment demonstrated the following key contributions:

- Effective Fault Masking: During a targeted Stealthy Delay Attack on one of the three timing sources (GMi), the system successfully identified and rejected the compromised signal, allowing the system to continue operating with validated, accurate time. The final time output, Tfinal, remained consistently within the strict TVE 1 % tolerance of the IEEE C37.118 standard. This confirmed the algorithm's ability to continuously deliver accurate time despite an active cyberattack (Case II).
- Total Failure Protection: In a scenario involving simultaneous attacks on two independent timing sources (Case III), the algorithm correctly detected the loss of consensus, triggered a critical alarm, and prevented the output of a distorted time. The immediate transition to internal Holdover mode ensured that PMUs did not receive a faulty signal, thereby avoiding system instability.
- Low Masking Latency: Our system demonstrated minimal latency between fault detection and the stabilization of the Tfinal output (e.g., less than 3 PTP cycles), which is crucial for real-time applications.

This work confirms that architectural redundancy combined with an intelligent consensus algorithm is the most effective strategy for protecting critical time references in digital substations. The proposed TMR architecture offers a model that significantly enhances the resilience of infrastructure against Advanced Persistent Threats (APTs).

ACKNOWLEDGMENT

This work was supported by the Ministry of Education, Science, Research and Sport of the Slovak Republic and the Slovak Academy of Sciences under contract no. VEGA 1/0627/24, 1/0647/26, Research on the impact of Vehicle-to-Grid (V2G) technology on the stability and flexibility of electrical grids using advanced simulations under contract no. 01/TUKE/2026.

REFERENCES

[1] R. Štefko, K. Eliáš, K. Glajc, A. Hyseni, F. Margita and J. Šimčák, "Cybersecurity Challenges in the Power Sector: Analysing Attacks on Electrical Grids and Substations," *2025 IEEE 23rd World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, Stará Lesná, Slovakia, 2025, pp. 000459-000464, doi: 10.1109/SAMI63904.2025.10883298.

[2] U.S. Department of Energy (DOE), "Securing Wide Area Measurement Systems," 2011. [Online]. Available: https://energy.gov/sites/prod/files/oeprod/DocumentsandMedia/8-Securing_WAMS.pdf.

[3] NERC Reliability Guideline, "PMU Placement and Installation," 2016. [Online]. Available: https://www.nerc.com/comm/RSTC_Reliability_Guidelines/Reliability%20Guideline%20-%20PMU%20Placement.pdf

[4] I. Macola, "The five worst cyberattacks against the power industry since 2014," Power Technology, 2020. [Online]. Available: https://www.power-technology.com/features/the-five-worst-cyberattacks-against-the-power-industry-since2014/?cf-view.

[5] A. Waleed, and M. Schukat. 2022. "A Security Enhancement of the Precision Time Protocol Using a Trusted Supervisor Node" Sensors 22, no. 10: 3671. https://doi.org/10.3390/s22103671.

[6] R. Stefko, M. Bobeck and Z. Conka, "Research of WAMS in Power Systems using SCADA System," 2023 IEEE 6th International Conference and Workshop Óbuda on Electrical and Power Engineering (CANDO-EPE), Budapest, Hungary, 2023, pp. 000053-000058, doi: 10.1109/CANDO-EPE60507.2023.10418038.

[7] M. Todescato, R. Carli, L. Schenato, and G. Barchi. 2020. "Smart Grid State Estimation with PMUs Time Synchronization Errors" Energies 13, no. 19: 5148. https://doi.org/10.3390/en13195148.

[8] E. Shereen, "Security of Time Synchronization for PMU-based Power System State Estimation: Vulnerabilities and Countermeasures," DiVA portal, 2021. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:1607196/FULLTEXT01.pdf

[9] A. Sundararajan, T. Khan, A. Moghadasi and A. I. Sarwat, "Survey on synchrophasor data quality and cybersecurity challenges, and evaluation of their interdependencies," in Journal of Modern Power Systems and Clean Energy, vol. 7, no. 3, pp. 449-467, May 2019, doi: 10.1007/s40565-018-0473-6.

[10] M. Agustoni, P. Castello, G. Frigo, and G. Gallus. 2023. "Time Synchronization Sensitivity in SV-based PMU Consistency Assessment" Metrology 3, no. 1: 99-112. https://doi.org/10.3390/metrology3010006.

[11] Y. Weng and Y. Zhang. 2023. "A Survey of Secure Time Synchronization" Applied Sciences 13, no. 6: 3923. https://doi.org/10.3390/app13063923.

[12] W. Alghamdi, and M. Schukat. 2020. "Cyber Attacks on Precision Time Protocol Networks—A Case Study" Electronics 9, no. 9: 1398. https://doi.org/10.3390/electronics9091398.

[13] R. Khan, K. McLaughlin, D. Laverty and S. Sezer, "Analysis of IEEE C37.118 and IEC 61850-90-5 synchrophasor communication frameworks," 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, MA, USA, 2016, pp. 1-5, doi: 10.1109/PESGM.2016.7741343.

[14] R. Styles, J. Asiamah, R.B. Hink, J. Schibonski, A. Werth, G. Hahn, E. Piesciorovsky, A. Lee, "Cyber Resilience in the CAST Timing System," 2024. [Online]. Available: https://info.ornl.gov/sites/publications/Files/Pub210822.pdf.

[15] R. Khan, K. Mclaughlin, D.M. Laverty and S. Sezer. "IEEE C37.118-2 Synchrophasor Communication Framework - Overview, Cyber Vulnerabilities Analysis and Performance Evaluation." International Conference on Information Systems Security and Privacy (2016).

[16] D.W.M. Piffaretti, A.F.P. Fernande s and G.M.S. Dias, "Cybersecurity analysis on precision time protocol," Research Square, 2024. [Online]. Available: https://www.researchsquare.com/article/rs-3798024/v1.

[17] G. Antonova, "Do we have time to fix the time?" *PAC World*, 2020. [Online]. Available: https://www.pacw.org/do-we-have-time-to-fix-the-time.

[18] F. Steinhauser, F. Fischer and W. Sattinger, "Synchrophasors, PMUs, and WAMS – Definitions and Testing," PAC World, 2025. [Online]. Available: https://www.pacw.org/synchrophasors-pmus-and-wams-definitions-and-testing.

[19] P. Castello, C. Muscas, P. A. Pegoraro and S. Sulis, "Automated test system to assess reporting latency in PMUs," 2016 IEEE International Instrumentation and Measurement Technology Conference Proceedings, Taipei, Taiwan, 2016, pp. 1-6, doi: 10.1109/I2MTC.2016.7520346.

[20] J. Lázaro, A. Astarloa, M. Rodríguez, U. Bidarte, and J. Jiménez. 2021. "A Survey on Vulnerabilities and Countermeasures in the Communications of the Smart Grid" Electronics 10, no. 16: 1881. https://doi.org/10.3390/electronics10161881.

[21] I. Diahovchenko, I. Yevtushenko, M. Kolcun, Z. Čonka, T. Zahorodnia, P. Vasyleha, 'Demand-Supply Balancing in Energy Systems with High Photovoltaic Penetration, using Flexibility of Nuclear Power Plants', Acta Politechnica Hungarica, vol. 20, no. 11, p. 115-135, 2023, doi: 10.12700/APH.20.11.2023.11.8.

[22] M. Bobček, R. Štefko, Z. Čonka, 'Electrical Protection Systems for the Evolving Microgrid Environment', Acta Politechnica Hungarica, vol. 20, no. 11, p. 159-178, 2023, doi: 10.12700/APH.20.11.2023.11.8.

[23] A.B. Asghar, K. Naveed, A. A. Aly, B. Alamri and R. Štefko, 'Estimation of Wake Effect in Wind Farms, using Machine Learning Algorithms', Acta Politechnica Hungarica, vol. 22, no. 11, p. 161-181, 2025, doi: 10.12700/APH.22.11.2025.11.10.

M. Beluscak, R. Merges, M. Galbavy, R. Stefko, M. Bobcek, R.B. Mónika, Z. Conka, 'Enhanced Fault Detection and Accelerated Switching Methodology in Power Systems Utilizing GOOSE Messaging: Centralized Control Switch Acceleration (CCSA)', Acta Politechnica Hungarica, vol. 22, no. 11, p. 183-201, 2025, doi: 10.12700/APH.22.11.2025.11.11.

ADRESY AUTOROV

Vladimír Szomosi, Technická Univerzita v Košiciach, Katedra elektroenergetiky, Mäsiarska 74, Košice, 04001, Slovenská Republika, vladimir.szomosi@tuke.sk

Marek Bobček, Technická Univerzita v Košiciach, Katedra elektroenergetiky, Mäsiarska 74, Košice, 04001, Slovenská Republika, marek.bobcek@tuke.sk

Július Šimčák, Technická Univerzita v Košiciach, Katedra elektroenergetiky, Mäsiarska 74, Košice, 04001, Slovenská Republika, julius.simcak@tuke.sk

Róbert Štefko, Technická Univerzita v Košiciach, Katedra elektroenergetiky, Mäsiarska 74, Košice, 04001, Slovenská Republika, robert.stefko@tuke.sk

Zsolt Čonka, Technická Univerzita v Košiciach, Katedra elektroenergetiky, Mäsiarska 74, Košice, 04001, Slovenská Republika, zsolt.conka@tuke.sk