A Physically Unclonable Function Authentication Protocol to Secure IEEE C37.118 Communications

Taylah Griffiths[1], Mohiuddin Ahmed[1,2], Chadni Islam[1] and Paul Haskell-Dowland[1]
[1]School of Science, Edith Cowan University, Australia
[2]School of Computer Science and Information Technology, Adelaide University, Australia

t.griffiths@ecu.edu.au
m.ahmed.au@ieee.org
c.islam@ecu.edu.au
p.haskelldowland@ecu.edu.au

Abstract: This paper outlines the need for security on the smart grid, specifically for Phasor Measurement Units (PMUs) and Phasor Data Concentrators (PDCs). These devices are targeted due to their criticality to the smart grid and the insecure communication protocols used. Common attacks on these devices target the data sent between them. As such, integrity-based attacks are the focus of this paper, more specifically on False Data Injection attacks (FDIAs) and Time Synchronisation Attacks (TSAs). Security solutions are proposed throughout literature from mitigation and protection techniques to detection and resolution techniques. To secure the PMU and PDC, this paper proposes an authentication protocol, utilising Physically Unclonable Functions (PUFs). PUFs are utilised to ensure device-specific authentication, in addition to the use of other cryptographic techniques including fuzzy extractors, nonces, encryption algorithms, and identifiers. A testbed was developed, on which the protocol was implemented and tested against the integrity-based attacks, namely FDIA and TSA. Informal and formal security analyses were performed on the protocol, finding that the protocol was secure against multiple attacks, including the FDIA and TSA, and implemented mutual authentication and forward secrecy. The formal security analysis was implemented with Proverif, a software used to prove the security of authentication protocols. The informal security analysis is provided as a logical sequence to prove the protocol's security against attacks and inclusion of security proofs. A performance analysis was performed to ensure the protocol had low enough computation, communication, and storage overheads.

Keywords: Authentication, Phasor measurement unit, Physically unclonable function, Phasor data concentrator, False data injection, Time synchronisation attack

1. Introduction

The International Energy Agency reports that the transmission sector of the smart grid grew by 10% in 2023 globally. The transmission sector is responsible for grid monitoring, transforming, and transmitting energy (IEA, 2025). Phasor Measurement Units (PMUs) are vital to grid stability as the devices provide monitoring of the grid and calculate an estimation of the state of the grid. The data includes current, voltage, frequency, and rate of change of frequency, also known as synchrophasor data. These data are passed to another device, the Phasor Data Concentrator (PDC), which checks the integrity of the synchrophasor packets and aligns the packets via timestamp, vital for state estimation (Paramo et al., 2022). The PMU and PDC are connected within the smart grid for bidirectional communication, making up the Wide Area Management System (WAMS) (Paramo et al., 2022), pictured in Figure 1. The PMU and PDC can communicate through different protocols, the main one being a plain-text protocol, IEEE C37.118 synchrophasor protocol. Due to the nature of plain-text protocols, devices communicating become susceptible to multiple vulnerabilities, for instance cyber-attacks (Cheng et al., 2023).

Common attacks targeting message integrity include data spoofing, injection, and replay attacks (Hasan et al., 2024). Confidentiality and availability are targeted with attacks such as Man-In-The-Middle (MITM) and Denial of Service (DoS). Other attacks that can impact the WAMS include Time Synchronisation Attacks (TSAs) and physical attacks (Nazir et al., 2025). False Data Injection Attacks (FDIAs) and TSAs are two attacks that can cause large impacts on the system (Sanati and Kamwa, 2023). Due to these attacks, security solutions are proposed in literature to safeguard systems. These solutions include authentication, encryption, key distribution schemes (KDSs), and digital signatures. These solutions focus on creating keys, identities, and cipher-text to mitigate attacks (Anantharaman et al., 2018, Hussain et al., 2022). Machine learning (ML) is another solution proposed for mitigation of cyber-attacks but can also be utilised for detection. ML techniques have been used to improve intrusion detection systems, existing authentication systems, and securing the data collected from smart grid devices (Nayak et al., 2023).

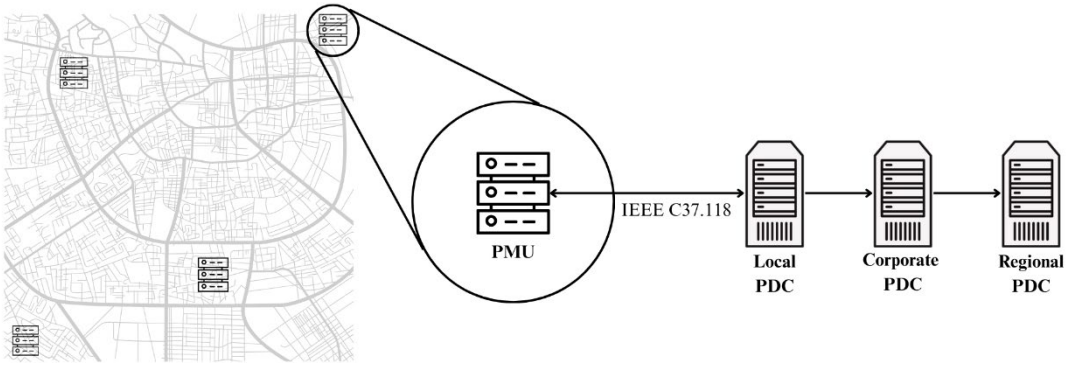

Figure 1: Basic WAMS layout demonstrating the communication between devices

Existing literature proposed solutions to secure the communication between PMU and PDC and its data. These solutions include security filters (Anantharaman et al., 2018), KDSs (Hussain et al., 2022, Farooq et al., 2023), ML (He et al., 2022), and data source authentication (Cui et al., 2021). All these solutions focus on securing the devices in different ways, for example, KDSs create a shared session key between devices. Security filters focus on detecting attacks within the network by searching the communications. Data source authentication methods detect spoofing attacks by checking the data for any modifications.

The goal of this paper is to propose an authentication protocol that can protect PMU and PDC communication against attacks, while achieving mutual authentication and creating a shared session key. The aim of the protocol is to specifically secure against FDIAs and TSAs since they can cause a large impact on the system. The authentication will prevent the FDIAs and TSAs while keeping the communication fast and reliable. The main contribution of this paper is an authentication protocol and analyses, including informal, formal, and performance. The protocol utilises Physically Unclonable Functions (PUFs) amongst other cryptographic techniques to achieve mutual authentication. A weak and a strong PUF are utilised as well as encryption, hash functions, XORs, nonces, and timestamps. The informal security analysis reveals the protocol to mitigate against, FDIA, TSA, replay, DoS, and physical attacks. The formal security analysis is performed with Proverif, indicating the protocol is secure. The performance analysis calculates the performance, outlining low computation, and small communication and storage overheads. The rest of this paper is formatted as follows: Section 2 covers the related works; Section 3 outlines the background; Section 4 outlines the proposed protocol; Section 5 describes the implementation of the protocol on a testbed; Section 6 provides the analyses; and Section 7 concludes this paper.

2. Existing Literature

Some existing works propose authentication protocols and other techniques for securing PMUs and WAMS. Anantharaman et al. (2018), created a security filter for IEEE C37.118 to parse frames to distinguish between the current message and previous messages. The purpose of their filter, PhasorSec, is to identify and remove any packets that may have been compromised. Their work mentions the fact that FDIAs are not protected against with PhasorSec, although their protocol could be extended with other techniques, to protect against such attacks.

Two papers create security solutions like KDSs to secure the PMU data when communicating between devices. Message authentication codes and certificate authorities are utilised in one scheme, testing the scheme on an experimental setup to find the size of the synchrophasor packet is impacted. This scheme is created to mitigate against attacks on IEEE C37.118 (Hussain et al., 2022). The following scheme is created to secure data and authenticate devices using the IEC 61850-90-5 communication protocol. This scheme is tested against TLS 1.2 and 1.3, finding 1.3 to be more efficient (Farooq et al., 2023). Neither of these papers record attacks secured by the proposed schemes. Two papers utilised existing techniques to secure PMU data and communications. Qiu et al. (2012) use cryptographic algorithms to secure communications of PMU and smart meters (SMs). While He et al. (2022) use existing ML methods to authenticate PMU data. They found that Continuous Wavelet Transforms with Convolution Neural Network performed the best. Javed et al. (2024) proposed a data provenance protocol that uses PUFs and quantum principles. The attack focused on within the security analysis was MITM attack. Hussain (2026) created a security scheme that makes use of digital signatures across IEEE C37.118. Performance analysis is calculated; however, a full security analysis is not performed. Table 1 outlines

the above papers, the techniques used to secure PMUs, the attacks mentioned within the paper to be secured, and the security proof claimed within the paper.

Table 1: Proposed authentication protocols in recent literature

Reference	Techniques	Attacks	Proofs
(Anantharaman et al., 2018)	Frame Filter	AFL Fuzzer	N/A
(Hussain et al., 2022)	Certificates, Key Distribution, Encryption, Hash Functions	N/A	Confidentiality, Integrity
(Qiu et al., 2012)	Encryption	N/A	Increased Energy Efficiency
(Farooq et al., 2023)	Key Distribution, Certificates, Encryption	N/A	Improved Security, Performance
(He et al., 2022)	Machine Learning	N/A	Accuracy, Predict Time, Pre-process Time
(Javed et al., 2024)	Hash Functions, PUFs, Quantum Unreality, Quantum Uncertainty	MITM	Privacy, Unpredictability
(Hussain, 2026)	Signatures, Hash Functions, Encryption	N/A	Message Integrity
This paper	Hash Functions, XOR PUFs, Encryption, Fuzzy Extractor	FDIA, TSA, Replay, DoS, Physical	Mutual Authentication, Forward Secrecy

The following papers all propose methods for synchrophasor data source authentication. The algorithms are created to protect against data spoofing attacks by detecting modifications in the data source information. Different ML and deep learning algorithms are utilised to achieve the data source authentication. Existing methods include “Mathematical Morphological Decomposition and Multi-Weighted Deep Stacking Forest” (Cui et al., 2022), “Multiscale Adaptive Coupling Correlation Detrended Analysis” (Bai et al., 2023), Quadratic Kernel Support Vector Machine (Liu et al., 2022b), “Self-Adaptive Mathematical Morphology and Time-Frequency” (Cui et al., 2021). While these papers help to secure the PMU data, they are not achieving the same communication security as proposed in this paper.

3. Background

The smart grid consists of different systems to generate, distribute, and transmit energy and information. There are two main systems for monitoring the energy distributed across the grid and they are Supervisory Control and Data Acquisition (SCADA) system and WAMS. The SCADA system utilises devices such as intelligent electronic devices or remote terminal units to gather grid data. The SCADA reports at a rate of every two to five seconds, reporting active/reactive power information such as power injections and flows, they also report on voltage magnitudes. The WAMS monitors the voltage, current, frequency, and rate of change of frequency, and consists of PMUs and PDCs. The PMUs report anywhere between five to 60 data scans every second. The PMUs capture this information from the grid and pass this data onto the PDC. The PDCs then collate this data, ordered by time, and either analyse the data or transfer it onto the next PDC. Some PDCs have the capability to analyse the synchrophasor data, while others simply collate, align, and store the information (Cheng et al., 2023). The communication focus of this paper is between the PMU and PDC.

3.1 Communication of PMU and PDC

There are two main protocols that have been used for synchrophasor communication, the first was IEEE 1344, which was replaced by IEEE C37.118 in 2011 (Grewal et al., 2014). The IEEE C37.118 protocol is the focus for this paper due to its popularity. The protocol uses TCP as a base, consists of five fields within the header, and has three frame types. The configuration frame is sent at the beginning of communication between the PMU and PDC to outline the structure of the data frame. The command frame contains information instructing the PMU of an action, for example, sending configuration frame or starting data transmission. The data frame has seven fields and sends the data collected from the grid. The data frame is the main frame needing to be secured as it is transferring energy information in plain-text across the network. Figure 2 shows the fields of the IEEE C37.118 protocol.

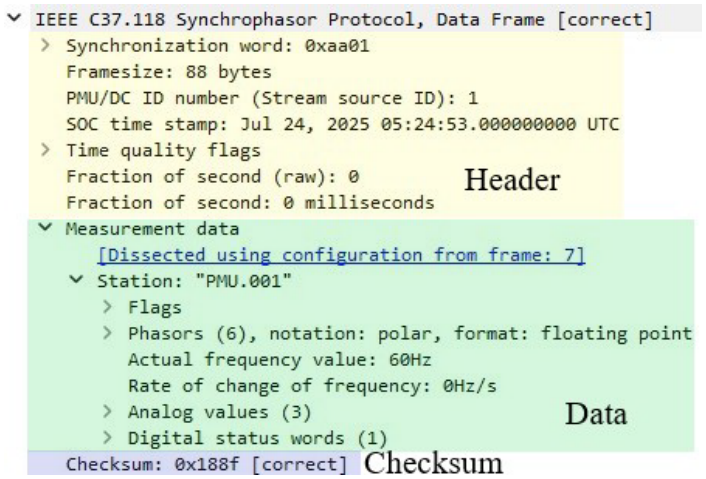

Figure 2: Wireshark capture of IEEE C37.118 data frame

The IEEE C37.118 communication protocol is wrapped with TCP/IP, thus when the PMU sends synchrophasor data the PDC sends back an acknowledgment.

3.2 Vulnerabilities of IEEE C37.118

An adversary that gains access to the smart grid network, can view the synchrophasor data being sent across the network. Many attacks can be targeted towards the communication between the PMU and PDC, impacting the confidentiality, integrity, or availability of the grid. Confidentiality is impacted when the adversary gains access to the network and views the data being sent, for example, MITM. Integrity is impacted when the data being passed across the network is modified or removed, such as a FDIA (Sanati and Kamwa, 2023). Availability is impacted when the system is shut down, such as a DOS. The most common attacks mentioned in recent literature includes, MITM, FDIA, spoofing, DOS, and replay attacks (Amanlou et al., 2025). The attack focus of this research is FDIAs and TSAs. TSAs can impact the performance of the system and cause incorrect actions to be taken by misguiding operators (Zhang et al., 2013). FDIAs situational awareness of the smart grid through modifying PMU data measurements (Sanati and Kamwa, 2023).

4. Proposed Protocol

This section outlines the authentication protocol, describing the components, and the different phases.

4.1 Preliminaries

4.1.1 Physically unclonable functions

PUFs are typically sorted into two categories, strong and weak. A strong PUF is defined as a security parameter that can grow to be quite large. If the PUF does not have this capability, then it is considered weak (Armknecht et al., 2011). Weak PUFs are typically used as keys while strong PUFs used for authentication. However, strong PUFs are susceptible to ML attacks, while weak PUFs are not (Santiago et al., 2017). Therefore, for the proposed protocol, we utilise both PUFs. The protocol makes use of one weak PUF, PUF_W and one strong PUF, PUF_S. The PUF_W is used as the key for encryption during the authentication phase of the protocol. PUF_S and PUF_W are used as part of the session key for encryption once the PMU and PDC have been authenticated. The PUFs require a challenge to output a response, the equation is written as: $PUF_X(Ch_X) = R_X$.

4.1.2 Fuzzy extractor

The protocol uses fuzzy extractors to ensure the noise from the response does not impact the protocol. The fuzzy extractor uses generation and reproduction algorithms. Generation is run first to get the helper data, hd_X, and the key, K_X. The generation algorithm for the fuzzy extractor is written as: $FE.Gen(R_X) = K_X, hd_X$. The reproduction algorithm is run each time a stored response is retrieved or regenerated to dissipate the noise and get the K_X. The reproduction algorithm is written as: $FE.Rep(hd_X, R_X) = K_X$.

4.1.3 Other cryptographic primitives

The protocol employs other cryptographic techniques to strengthen the security. These techniques include, hash functions, XORs, nonces, and timestamps. Timestamps are used to ensure the messages do not go through

another path on their transit to the other device. Hash functions, XORs, and nonces are used to make the messages difficult to interpret by an adversary. The protocol also utilises an encryption algorithm to securely send data between the PMU and PDC.

4.2 Authentication Protocol Phases

The proposed authentication protocol is made up of two phases, namely, initialization and authentication. The initialization phase generates initial challenge, response, and helper data to be stored with the PDC and PMU. The authentication phase allows the devices to create a shared session key allowing for encryption.

4.2.1 Initialisation phase

This phase sends the challenge to the PMU through a secure channel, with the response used with the fuzzy extractor generation algorithm to store the helper data at the PMU. The PMU ID, response, challenge, and helper data are all sent to the PDC through a secure channel. Figure 3 shows the steps for the initialisation phase.

Algorithm 1 – Name of Protocol – Initialisation Phase
PDC_x:
Challenges, Ch_s and Ch_w are generated.
Challenges are sent to the PMU via a secure channel.
PMU_y:
Strong response is calculated $R_s = PUF(Ch_s)$
Strong helper data, hd_s, and key is calculated, $FE.Gen(R_s)$
Weak response is calculated $R_w = PUF(Ch_w)$
Weak helper data, hd_w, and key is calculated, $FE.Gen(R_w)$
PMU retrieves its own ID, ID_y.
Helper data is stored.
Challenges, responses, helper data, and ID is sent to the PDC via a secure channel.
PDC_x:
Challenges, responses, helper data, and ID is stored.

Figure 3: Algorithm of initialisation phase of proposed protocol

4.2.2 Authentication phase

The authentication phase is initiated every time communication between the PMU and PDC is requested. This phase results in a shared session key between the PMU and PDC allowing for the PMU data to be encrypted. To mutually authenticate, verifiers are used to confirm the identity of each device, while timestamps are used to ensure the communication has not been intercepted on transit. The session key is created from components used throughout the phase. Figure 4 shows the steps for the authentication phase.

5. Implementation

The testing of the proposed authentication protocol was performed by implementing the protocol on a testbed. This section discusses the testbed, the attacks on the testbed, and the protocol performance.

5.1 Testbed

The testbed was formed to view, attack, and protect the IEEE C37.118 protocol. The PMU and PDC are formed from simulators, on a Raspberry Pi (RPi) and Dell PC, respectively, and communicate using a wired connection through a router. Other components include two other RPis that launch an attacker and a time server. The first stage was to set up the communication between the PMU and PDC, ensuring the two devices were communicating via IEEE C37.118. Once this communication was established, the time server was added to grab the time via GPS and pass that time onto the PMU and PDC devices through the Network Time Protocol (NTP).

Algorithm 2 – Name of Protocol – Authentication Phase	
PDC_x	**PMU_y**
Timestamp, T_1, and nonce, r_1, are generated.	
Challenges, Ch_w, Ch_s, Helper data, hd_w, hd_s, Responses, R_w, R_s, and ID_p are retrieved from storage.	Timestamp, T_2, and nonce, r_2, are generated.
	Helper data, hd_w, hd_s, retrieved from storage.
Nonce share, N_1 is calculated by $ID_p \oplus r_1$	ID_p is retrieved and hashed, ID_h
Key, K_w, is produced with the fuzzy extractor, $FE.Rep(hd_w, R_w) = K_w$	$ID_h \oplus H_1$ to get Ch_w
	Response, R_w is calculated $PUF(Ch_w)$
Message, M_1, is created with the encryption algorithm and K_w, $E_{Kw}(T_1 \mid N_1 \mid Ch_s)$	Key, K_w is calculated with $FE.Rep(hd_w, R_w)$
	M_1 is decrypted, $D_{Kw}(M_1)$ to get T_1, N_1, Ch_s
Hashed message, H_1, created by $h(ID_p) \oplus Ch_w$	T_1 is checked against T_2, if too far apart, messages may be intercepted, communication terminated.
M_1 and H_1 are sent to the PMU_y	
	Response, R_s is calculated $PUF(Ch_s)$
	Key, K_s is calculated with $FE.Rep(hd_s, R_s)$
	Nonce, r_1, is retrieved by $ID_p \oplus N_1$
	Verifier 1, V_1, is formed with $h(K_s \mid ID_p \mid r_1)$
	Message 2, M_2, is created with the encryption algorithm and K_w, $E_{Kw}(r_2 \oplus ID_p \mid V_1 \mid T_2)$
	M_2 is sent to the PDC_x
Timestamp, T_3, is generated.	
M_2 is decrypted, $D_{Kw}(M_2)$ to get T_2, V_1, H_2	
T_2 is checked against T_3, if too far apart, messages may be intercepted, communication terminated.	
	Timestamp, T_4, is generated.
Key, K_s is calculated with $FE.Rep(hd_s, R_s)$	M_3 is decrypted, $D_{Kw}(M_3)$ to get N_2, T_3
V_1, is verified by PDC_x calculating $V_1' = h(K_s \mid ID_p \mid r_1)$ and comparing V_1 and V_1'	T_3 is checked against T_4, if too far apart, messages may be intercepted, communication terminated.
H_2 is xored with ID_p to get nonce, r_2	
Verifier 2, V_2, is formed with $h(r_2 \mid ID_p \mid r_1)$	$N_2 \oplus r_2$ to get V_2
Message 3, M_3, is created with the encryption algorithm and K_w, $E_{Kw}(V_2 \oplus ID_p \mid T_3)$	V_2, is verified by PDC_x calculating $V_2' = h(r_2 \mid ID_p \mid r_1)$ and comparing V_2 and V_2'
Session Key, SK, is created $h(r_2 \mid ID_p \mid K_w \mid K_s)$	Session Key, SK, is created $h(r_2 \mid ID_p \mid K_w \mid K_s)$
Message 4 is created to obtain new strong challenge, $E_{SK}(Ch_{sx})$	M_4 is decrypted, $D_{SK}(M_4)$ to get Ch_{sx}
	New s-response is calculated $R_{sx} = PUF(Ch_{sx})$
M_3, M_4 are sent to the PMU_y	New helper data, hd_{sx}, and key is calculated, $FE.Gen(R_{sx})$
	M_5 is created with SK, $E_{SK}((R_{sx}, hd_{sx}) \oplus r_1)$
	M_5 is sent to PDC_x
Ch_{sx}, R_{sx}, hd_{sx} are stored.	

Figure 4: Algorithm of authentication phase of proposed protocol

5.2 Attacks

The attacks on the testbed were set up using a Python script to intercept the plain-text traffic. The attacks that were chosen were FDIA and TSA. The first attack to be implemented was the TSA. A Python script for NTP spoofing was retrieved online from GitHub. This script was modified to work with the devices and the testbed network. This attack targeted the communication between the time server and the PMU, to modify the PMU's time. Modifying the time incorrectly ordered the measurements when collated by the PDC. The FDIA was next, and this was implemented by modifying the script for the NTP spoofing attack. The attack targeted the plain-text within the IEEE C37.118 protocol to modify the frequency that was recorded by the PMU. The frequency was modified in between the PMU and PDC, so the PMU reported the correct value, but the incorrect value reached the PDC.

5.3 Experimental Analysis

The protocol was developed in Python, with scripts in between the two simulators. The scripts would listen for the devices and when they connected the authentication protocol would be run. If the protocol was successful in mutual authentication then the simulators could connect, and the data was encrypted with the shared session key. If the verification was not mutually successful, the script would terminate communication. The FDIA was unable to be executed as the synchrophasor data was encrypted. The TSA failed as the authentication protocol checks timestamps, if the PMU time was changed the authentication protocol would terminate.

6. Protocol Analysis

6.1 Informal Security Analysis

6.1.1 False data injection attacks

The objective of the adversary in a FDIA is to modify data as it is passed across the network to make the receiving device misjudge the data. Since the PMU and PDC communicate bi-directionally, the adversary can target either communication from PDC to PMU or PMU to PDC. However, the protocol utilises encryption and hash functions, so if the adversary modifies the data being sent, the messages will not be decrypted properly. If the verifier gets modified the protocol will end communication since the verifiers do not match. Thus, making the FDIA unsuccessful.

6.1.2 Time synchronisation attacks

The objective of the adversary in a TSA is to misguide communications and information by modifying the timestamp of a device. The synchronisation of PMUs is vital to measure phasor data across multiple PMUs. However, the authentication protocol uses timestamps to check the validity of the messages, so if the timestamps are modified, communication will be disconnected. Thus, making the TSA unsuccessful.

6.1.3 Other attacks

The objective of the adversary when executing a replay attack is to send previously sent messages to gain access to the device. However, the protocol implements the use of rotating keys and nonces, so if an old message is sent to either device, communication will be terminated based on incorrect data being received. Thus, making the replay attack unsuccessful. The objective of the adversary during a DoS attack is to send so many messages to an entity on a network that the computation power is overtaken, and they cannot receive the legitimate messages sent. Since the authentication protocol terminates communication if a message comes through that is incorrect, the DoS attack will be unsuccessful. The objective of the adversary in a physical attack is to physically gain access to the device through proximity rather than remotely. With the use of PUFs, the PMU is not at risk if physically apprehended since the PMU does not store the responses from the challenges. Thus, making the physical attack unsuccessful.

6.1.4 Security criteria

Mutual authentication is the process of the devices authenticating one another. This is vital for the PMU and PDC communication since they communicate to each other. The verifiers in the protocol allow for this. The verifier for the PMU is the hash of $h(K_S, ID_P, r_1)$, the key and nonce are different each time the protocol runs, so the verifier cannot be recorded. The verifier for the PDC is the hash of $h(r_2, ID_P, r_1)$, both nonces change each iteration of the protocol, so the verifier cannot be recorded. Forward secrecy means that if the session key is obtained by an adversary, it does not reveal anything about future session keys or is not a future session key. The session key of this protocol is the hash of $h(r_2, K_S, ID_P, K_W)$, the contents of the session key cannot be revealed as they are securely hashed; the nonce and keys are changed each iteration of the protocol so the session key cannot be recorded.

6.2 Formal Security Analysis

This section outlines the use of a software that confirms secrecy of protocol variables and verifies the protocol's ability to confirm values that are received. Proverif is based on the Dolev-Yao model that allows users to write out the logic for their authentication protocol to prove values are kept secret throughout communication. There were three variables we did not want the attacker to learn while the protocol was running. They were the ID and the two PUF responses, weak and strong. In the Proverif logic, we queried these variables to ensure their secrecy would not be impacted during the protocol's runtime. Proverif was also used to ensure the two devices authenticated one another, Firstly, in the logic, by ensuring the event of the PDC authentication occurred after the PMU started authentication (and vice versa); secondly, with a query to make sure that anytime the PDC accepts a message from the PMU there is a prior occurrence of the PMU sending the exact message (and vice versa). The Proverif logic was run to find that the attacker cannot learn the PMU ID, or responses of the PUFs, the main sources of secrecy. The other piece of information confirmed in Proverif was that the PMU and PDC authenticated one another properly, and the timestamps were accepted correctly. Figure 5 shows the verified outcome from Proverif.

```
------------------------------------------------------------
Verification summary:

Query not attacker(ID) is true.

Query not attacker(Rw[]) is true.

Query not attacker(Rs[]) is true.

Query inj-event(BAuthed) ==> inj-event(AStarted) is true.

Query inj-event(AAuthed) ==> inj-event(BStarted) is true.

Query inj-event(B_accepts_T(x)) ==> inj-event(A_sends_T(x)) is true.

Query inj-event(A_accepts_T(x)) ==> inj-event(B_sends_T(x)) is true.

------------------------------------------------------------
```

Figure 5: Verification summary from Proverif

6.3 Performance Analysis

6.3.1 Execution time

The computation time was recorded within the Python script, however due to Python's low computational power, these numbers were heightened significantly. The cryptographic primitives such as XOR and hash functions are recorded normally, but the cryptographic algorithms such as fuzzy extractor are much larger. Since these cryptographic algorithms are not recorded as in a real-world scenario, execution times as recorded by other papers are used (Liu et al., 2022a, Kaveh et al., 2023, Hussain et al., 2022). Figure 6 shows the final times as a combination of times recorded by Python for the primitives and normalised times based on other papers for the techniques.

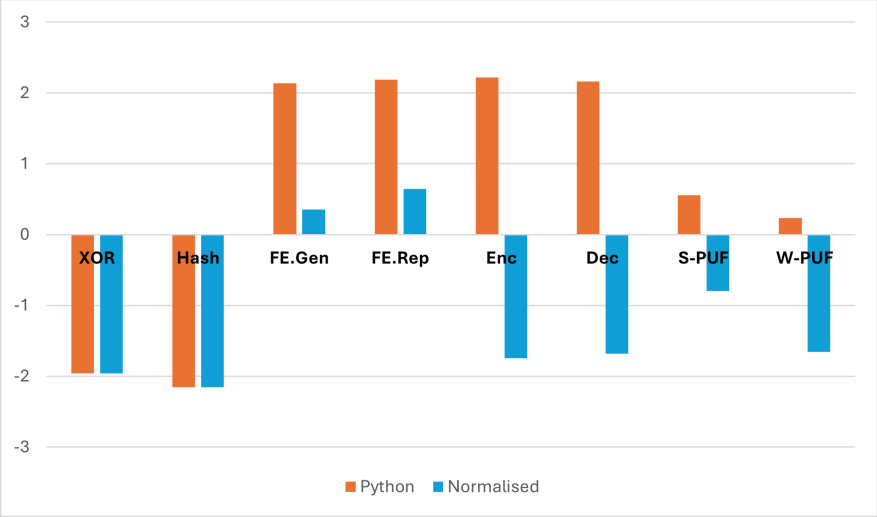

Figure 6: Logarithmic execution times of cryptographic techniques, Python vs normalised

6.3.2 Communication overhead

The communication measurement was calculated based on the number of bits sent over the course of the protocol. Figure 7 outlines the bits for each message sent across the protocol. The total bits are calculated based on the size of the messages sent, calculated from the cryptographic primitive sizes. The total size for all messages totalled to 200 bytes.

6.3.3 Storage overhead

The storage overhead is only calculated for the PMU device, since the PMU is the storage constrained device. The only things the PMU needs to store is the helper data for the strong and weak PUFs. The helper data, amounts to 16 bits each, making the total storage cost for the PMU is 4 bytes.

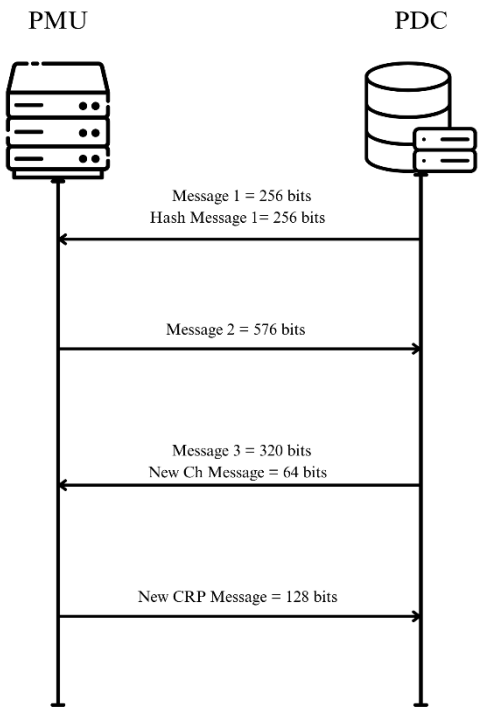

Figure 7: Messages in bits sent between devices

7. Conclusion

PMUs are vital devices for the smart grid to estimate the current state of the grid. They report the data of the grid to the PDC through a plain-text, insecure communication protocol. This makes the devices vulnerable to cyber-attacks, such as FDIAs and TSAs. This paper proposes an authentication protocol to protect the devices against such attacks. The protocol utilises PUFs to ensure the PMU does not have to store much data or use too much computational power. The protocol is analysed through a formal and informal security analysis and a performance analysis. It is found that the protocol is secure against multiple attacks, verifiably secure, and lower on computation, communication, and storage costs.

Acknowledgements

This research was supported by the Commonwealth through an Australian Government Research Training Program Scholarship.

Ethics declaration: Ethical clearance was out-of-scope as no humans, human data, or animals were used in the research.

AI declaration: AI was not used in the creation of this paper.

References

Amanlou, S., Hasan, M. K., Mokhtar, U. A., Malik, K. M., Islam, S., Khan, S., Khan, M. A. & Khan, M. A. 2025. Cybersecurity Challenges in Smart Grid Systems: Current and Emerging Attacks, Opportunities, and Recommendations. *IEEE Open Journal of the Communications Society,* 6**,** pp. 1965-1997.

Anantharaman, P., Palani, K., Brantley, R., Brown, G., Bratus, S. & Smith, S. W. PhasorSec: Protocol Security Filters for Wide Area Measurement Systems. 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2018. pp. 1-6.

Armknecht, F., Maes, R., Sadeghi, A. R., Standaert, F. X. & Wachsmann, C. A Formalization of the Security Features of Physical Functions. 2011 IEEE Symposium on Security and Privacy, 2011. pp. 397-412.

Bai, F., Cui, Y., Yan, R., Yin, H., Chen, T., Dart, D. & Yaghoobi, J. 2023. Cost-effective synchrophasor data source authentication based on multiscale adaptive coupling correlation detrended analysis. *International Journal of Electrical Power & Energy Systems,* 144**,** pp. 108606.

Cheng, G., Lin, Y., Abur, A., Gómez-Expósito, A. & Wu, W. 2023. A Survey of Power System State Estimation Using Multiple Data Sources: PMUs, SCADA, AMI, and Beyond. *IEEE Transactions on Smart Grid,* PP**,** pp. 1.

Cui, Y., Bai, F., Saha, T. & Yaghoobi, J. 2022. Authenticating source information of distribution synchrophasors at intra-state locations for cyber-physical resilient power networks. *International Journal of Electrical Power & Energy Systems,* 139**,** pp. 108009.

Cui, Y., Bai, F., Yan, R., Saha, T., Ko, R. K. L. & Liu, Y. 2021. Source Authentication of Distribution Synchrophasors for Cybersecurity of Microgrids. *IEEE Transactions on Smart Grid,* 12**,** pp. 4577-4580.

Farooq, S. M., Aftab, M. A., Hussain, S. M. S. & Ustun, T. S. A Key Distribution Scheme based on TLS 1.3 for IEC 61850-90-5 Synchrophasor Communication. 2023 IEEE International Conference on Power Electronics, Smart Grid, and Renewable Energy (PESGRE), 17-20 Dec. 2023 2023. pp. 1-4.

Grewal, S. K., Soni, M. K. & Jain, D. K. 2014. Requirements and challenges of PMUs communication in WAMS environment. *Far East Journal of Electronics and Communications,* 13**,** pp. 121-135.

Hasan, M. K., Abdulkadir, R. A., Islam, S., Gadekallu, T. R. & Safie, N. 2024. A review on machine learning techniques for secured cyber-physical systems in smart grid networks. *Energy Reports,* 11**,** pp. 1268-1290.

He, Q., Bai, F., Cui, Y. & Zillmann, M. Machine Learning-based Cybersecurity Defence of Wide-area Monitoring Systems. 2022 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia), 8-11 July 2022 2022. pp. 991-996.

Hussain, S. M. S. 2026. Public key-based quantum-resistant security scheme for IEEE C37.118.2 PMU communication. *Electric Power Systems Research,* 252**,** pp. 112378.

Hussain, S. M. S., Farooq, S. M. & Ustun, T. S. 2022. A Security Mechanism for IEEE C37.118.2 PMU Communication. *IEEE Transactions on Industrial Electronics,* 69**,** pp. 1053-1061.

IEA. 2025. *Building the Future Transmission Grid: Executive Summary* [Online]. International Energy Agency. Available: https://www.iea.org/reports/building-the-future-transmission-grid/executive-summary [Accessed November 2025].

Javed, K., Khan, M., Ullah, M., Aman, M. & Sikdar, B. 2024. Securing Synchrophasors Using Data Provenance in the Quantum Era. *IEEE Open Journal of the Communications Society,* 5**,** pp. 1594-1608.

Kaveh, M., Mosavi, M. R., Martín, D. & Aghapour, S. 2023. An efficient authentication protocol for smart grid communication based on on-chip-error-correcting physical unclonable function. *Sustainable Energy, Grids and Networks,* 36**,** pp. 101228.

Liu, F., Yan, Y., Sun, Y., Liu, J., Li, D. & Guan, Z. 2022a. Extremely Lightweight PUF-based Batch Authentication Protocol for End-Edge-Cloud Hierarchical Smart Grid. *Security and Communication Networks,* 2022**,** pp. 9774853.

Liu, S., You, S., Yin, H., Lin, Z., Liu, Y., Cui, Y., Yao, W. & Sundaresh, L. 2022b. Data source authentication for wide-area synchrophasor measurements based on spatial signature extraction and quadratic kernel SVM. *International Journal of Electrical Power & Energy Systems,* 140**,** pp. 108083.

Nayak, J., Al-Dabass, D., Pelusi, D. & Mishra, M. 2023. Special issue on machine learning for security and privacy: advancing the state-of-the-art applications. *Neural Computing and Applications,* 35**,** pp. 4809-4812.

Nazir, R., Laghari, A. A., Dahri, F. H., Shoulin, Y., Alhakeem, Z. M., Hakim, H. & Mughal, Z. A. 2025. A review on machine learning techniques for network security. *Journal of Cyber Security Technology***,** pp. 1-45.

Paramo, G., Bretas, A. & Meyn, S. 2022. Research Trends and Applications of PMUs. *Energies* [Online], 15.

Qiu, M., Su, H., Chen, M., Ming, Z. & Yang, L. T. 2012. Balance of security strength and energy for a PMU monitoring system in smart grid. *IEEE Communications Magazine,* 50**,** pp. 142-149.

Sanati, S. & Kamwa, I. A Comprehensive Review of Cyber Security Enhancements for PMU Communications in Microgrids. 2023 4th International Conference on Clean and Green Energy Engineering (CGEE), 2023 Ankara, Turkiye. pp. 11-18.

Santiago, L., Patil, V. C., Prado, C. B., Alves, T. A. O., Marzulo, L. A. J., França, F. M. G. & Kundu, S. Realizing strong PUF from weak PUF via neural computing. 2017 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 23-25 Oct. 2017 2017. pp. 1-6.

Zhang, Z., Gong, S., Dimitrovski, A. D. & Li, H. 2013. Time Synchronization Attack in Smart Grid: Impact and Analysis. *IEEE Transactions on Smart Grid,* 4**,** pp. 87-98.