Energy Efficient Authentication Scheme for Industrial Smart Grid Environments

B. Informal security analysis

For ensuring the security of the proposed scheme, we used the collision-free one-way hash function and two hard problems: the ECDHP and CDHP, which we explained in Subsection II.*C*. In this subsection, we analyze and summarize the main security benefits of our proposed scheme. The proposed scheme is targeting optimized performance at a lower cost. Therefore, in this subsection, we present how the proposed scheme is secure against those known cryptographic attacks, which affect the performance of the authentication scheme. Furthermore, we compare the proposed scheme with the existing schemes by utilizing security attributes to explain its efficiency as follows:

1) Replay attacks: In this attack, an adversary may impersonate a legitimate device by reusing the message $< CID_i,\ X_i, Y_i >$ obtained from a previous protocol run and transfer it to the SP. After receiving the log-in request, SP computes and verifies whether $h(X_i.CK) \stackrel{?}{=} Y_i$ and sends back new challenges $< X_j, Y_j, T>$ to SM_i. However, after impersonating messages $<X_j, Y_j, T>$ send to SM_i , the attacker would be unable to compute $A = T \oplus I_i \oplus CK'$ without knowing I_i and CK'.

Notably, both CK' and I_i are neither sent through any messages over public channel nor can be acquired from the embedded device (SM_i) because of its tamper-proof design [29]. Thus, it cannot verify the challenge $C_i = H(Y_i||A)$. Without the knowledge of the secret key x_{i1} and x_{j1} of the device and server, the attacker cannot compute the valid session key $(S_k = x_{i1}.X_i.Psw_i.P_{ks} = x_{j1}.X_i.\mathrm{s}.P_{ke}$). Also, the security of the session key is well protected by ECDLP and CDHP. Therefore, it almost impossible to extract private key $<Psw_i, s>$ of the device, and the server in polynomial time. Hence, the proposed scheme is secure against replay attacks.

2) DoS attacks: To prevent the proposed scheme from DoS attack**,** the server (SP) terminates the login session if the number of incorrect attempts to enter CID_i reaches the maximum limit.

However, the login request will be continued as soon as the correct CID_i is entered. Furthermore, in the login phase, assume the adversary replaces message $<CID_i, X_i, Y_i >$ with $< CID_i', X_j', Y_j'>$ by randomly selecting the elliptical curve point x_{i1}' and sent it back to SP; however, the SP computes and compares the previous value with the received Y_i' $(\mathrm{i.e.}, h(X_i.CK)) \stackrel{?}{=} Y_i$. If SP finds a difference between both the values, it terminates the protocol with a failure message to the user. Therefore, the proposed scheme is secure for DoS attacks.

The resource optimization of the SG system can be influenced by the success of a major attack (i.e., DOS and Replay attack) [34]. The proposed scheme is, however, protective against both attacks. Therefore, the proposed scheme ensures the security of the performance of the proposed scheme. In addition, we have mentioned a couple of security attacks that can affect the performance of the authentication scheme at certain extent as follow:

3) Password-guessing attacks: In the proposed scheme, an embedded device password psw_i store in the form of a password generator (i.e., public key $(E_{ke} = psw_i.G)$) and wrapped in the form of $I_i = H(ID_i||psw_i)$. Consequently, the attacker cannot guess the password psw_i without knowing I_i and P_{ke}. Therefore, the proposed scheme maintains the security of the password by using ECDLP and a hash function ($h()$). Notably, I_i of SM_i is neither sent through any messages over open public channel nor can be acquired from the embedded device (SM_i) because of its tamper-proof design (i.e., nor is it stored in the SM_i and SP). Therefore, the proposed scheme prevented the password-guessing attack.

4) Server Impersonation attacks: Assume the scenario, where the phase of authentication of SM_i is impersonated by an adversarial server. An adversarial server impersonates and receives the parameters $<CID_i$, Xi , $Yi>$ from SM_i. Then adversary server randomly choose parameters $<X_j', Y_j', T'>$ and send it back to the SM_i. After receiving parameters $<X_j', Y_j', T'>$, the SM_i computes factor and check $h(X_j'.\mathrm{A}) \stackrel{?}{=} Y_j'$. However, equivalence does come wrong. This is because of, an attacker randomly computes challenge T as $T'$and completely unaware about challenge A. In addition, attackers unable to compute a session key. This is due to attacker unable to extract private server key (s) or device password (psw_i) from past session key due to its security protected by ECDLP and CDHP. Further, the security of s is protected in server $(E_{ks} = \mathrm{s}.ID_{SP}.G)$ by ECDLP and CDHP. Therefore, the proposed scheme is secure against the server impersonation attack.

5) Server database stolen attacks: In this case, if an attacker makes a server database attack, however, an attacker is unsuccessful in breaking the table of the server database. Since the security of the server, the database is protected by an encryption key $E_{ks} = s.\ ID_{SP}.G$, whose security protected by ECDLP and CDHP. Therefore, an attacker unable to extract either s or ID_{SP} from E_{ks}. In addition, even if the attacker got s by any means, the attacker still unable to extract ID_{SP} from E_{ks}. Further, if encryption key of the server is compromised by somehow, however, attacker unable to extract security parameter $<T, A, E_t>$ from $<t', a', e_t'>$. This is due to parameters security is still protected by ID_{SP} and s. Therefore, the proposed scheme is preventive against the server database stolen attack.

6) Embedded device Impersonation attacks or **Key compromise impersonation attacks [35]:** In this case, if an attacker impersonates toward server (SP) as real embedded devices (SM_i) by replaying the previous intercept message. However, an attacker still lacking SM_i secrete parameters $< psw_i, I_i, CK' >$. This is due to, that secrete parameter is protected due to tamper-proof design of SM_i. In addition, psw_i of SM_i in $I_i = H(ID_i||psw_i)$ and in $E_{ke} = psw_i.G$ is

protected by hash function $h()$ and ECDLP, respectively. Therefore, attacker unable to compute $A=T \oplus I_i \oplus CK'$ correctly without knowing parameters $< I_i, CK' >$. Result into incorrect deduction $h(X_j.\mathrm{A}) = ?\ Y_j$ which leads to the termination of the session. Further, an attacker is unsuccessful in extracting Psw_i from the past session key $(S_k = x_{i1}.X_i.Psw_i.P_{ks})$ whose security is protected by ECDLP and CDHP. Therefore, an attacker unable to compute a session key S_k for a current session without knowing Psw_i. Therefore, from the above reasons, an attacker to fail launch embedded device Impersonation attacks on SM_i.

7) Many logged-in users attacks: Suppose an adversary somehow managed to get a legally embedded device's credentials $< CID_i', CK' >$, along with the secret identity I_i. Subsequently, the adversary tries to communicate with the server by impersonating as SM_i. However, in the proposed scheme, out of all-knowing the valid credential, only one legal SM_i can communicate with SP at a time. As every time SP sets a working bit W_{bit-i} equal to one for the corresponding communication with SM_i after successful authentication and store the working bit in its database. Every time the receiver SP will check W_{bit-i} before establishing a connection with the requested SM_i. Furthermore, the receiver SP can deny all the requests if $W_{bit-i} = 1$ representing the existing SM_i is still communicating with it.

8) Server-spoofing attacks: In these attacks, an adversary may to masquerade as a server (SP) to know the secret credential of a device (SM_i). The SM_i's secrete credential CID_i comprises the hashing of some random secret (R), secret identity (I_i), and server secret (s). In addition, CID_i is stored in the SP encrypted database (i.e., $E_{ks} = s.\ ID_{SP}.G$) in the form of $CID_i' = CID_i.G$. Since security of E_{ks} is protected by ECDLP and CDHP. In addition, it is impossible for an attacker to extract to CID_i from $CID_i' = CID_i.G$. Thus, an attacker cannot get ED_i's secret credentials CID_i by any means due to the complexity of solving ECDLP and CDHP. Therefore, the proposed scheme is secure for server-spoofing attacks.

9) Forward secrecy attacks: Even if the private key of both the SM_i and the server (SP) is compromised by some other means, the confidentiality of the recently established session keys ought not to be affected. Suppose an adversary somehow discovers SM_i's password (Psw_i) and SP's secret key (s); thus, the adversary determines other components from the message. However, the adversary cannot derive the session key $(S_K = x_{i1}.X_j.Psw_i.P_{ks} = x_{j1}.\mathrm{X_i}.s.P_{ke})$. This is because to compute it, the adversary must determine x_{i1} and x_{j1} from X_i and X_j, which seems to be computationally infeasible because of the complexity of the ECDLP. Therefore, even if the present session key (S_K) is leaked, the adversary cannot determine all the past session keys, as the session key also depends upon the random secrets (x_i and x_j) whose security in (X_i and X_j) is protected by the concept of ECDLP and security in S_K is protected by the concept of ECDLP and CDHP. Hence, the proposed scheme is secure to forward secrecy attacks.

10) Insider attacks: In the proposed scheme, during the registration of a device, SM_i sends $I_i = H(ID_i||psw_i)$ instead of sending the password (psw_i), securely over the trusted channel (i.e., wolfSSL [33]). This process is done by the company, which we mentioned in registration subsection IV.*A*. After successful registration, the product is handover to the owner by the company. Since the owner can extract any data $< psw_i, I_i, CID_i, CK' >$ from device SM_i due to its tamper-proof design [29]. Thus, the advisor of SP cannot acquire the secret psw_i because it is ensured by

TABLE II
SECURITY ATTRIBUTES COMPARISON

Attacking scenarios	Zhou et. al. [16]	Yu et.al. [17]	Xie et.al. [18]	Chatterjee et.al. [20]	Yu et.al. [21]	Sengupta et.al. [22]	Yang et.al. [23]	Wang et.al. [24]	Wazid et.al. [25]	Proposed
Reply attack	n/a	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	**Supported**
Password guessing attack	Supported	n/a	n/a	n/a	n/a	Supported	n/a	Supported	Supported	**Supported**
Embedded device Impersonation attack	n/a	Supported	Supported		n/a	n/a	Supported	n/a	Supported	**Supported**
Denial of service attack	n/a	n/a	n/a	n/a	Supported	Supported	n/a	n/a	Supported	**Supported**
Many logged-in users attack	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	**Supported**
Server Impersonation attack	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	**Supported**
Forward secrecy	n/a	Supported	n/a	Supported	Supported	Supported	n/a	Supported	n/a	**Supported**
Insider attack	Supported	n/a	n/a	n/a	Supported	n/a	n/a	Supported	Supported	**Supported**
Known session - specific temporary information attack	n/a	n/a	n/a	Supported	n/a	n/a	n/a	Supported	n/a	**Supported**
Attack on user anonymity	Supported	Supported	n/a	n/a	Supported	Supported	Supported	Supported	Supported	**Supported**
Server database stolen attack	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	**Supported**
Cookie theft attack	n/a	n/a	Supported	n/a	n/a	n/a	n/a	Supported	n/a	**Supported**

SM_i's identity and $h()$. Thus, the privileged insider fails to impersonate the legitimate SM_i. Hence, the proposed scheme is secure against insider attacks.

11) Known session-specific temporary information attacks: After successful authentication, both the communicating SM_i and SP compute the session key $(S_K = x_{i1}.X_j.Psw_i.P_{ks} = x_{j1}.\mathrm{X_i}.s.P_{ke})$, whose security is protected by ephemeral secrets x_{i1} and x_{j1}. Suppose that an adversary somehow discovered the ephemeral secret x_{i1} or x_{j1}. However, the adversary cannot derive the session key (S_K) only with the knowledge of single ephemeral secrets. Since, the security of S_K still depend on Psw_i or s. Therefore, to derive the session key, the advisor must determine $<Psw_i, s>$ from the past session key (S_k), which, in turn, seems to be computationally infeasible because of the difficulties in solving the CDHP and ECDLP for pairs that are difficult to comprehend using a polynomial-time algorithm. Thus, the proposed scheme is securely infeasible to a known session-specific temporary information attack.

12) Attacks on user anonymity: Device anonymity implies that an attacker cannot discover the device's concealed identity (ID_i) by using the transmitted messages during the login and authentication phases. Here, a company who is selling device SM_i itself do the registration process and hide ID_i inside identity (I_i) using hash function h (). In addition, during the registration process, the company do the registration process with the server by sending $<I_i, psw_i>$ through a secure channel. Then, the company handover tamper-proof product to the owner. Therefore, the security of I_i is well protected by tamper-proof design of SM_i. Further, the security of I_i is well secured in CID_i using a secret key (s) and a random number (R) with the assistance of hash function $h()$. Moreover, both s and the R are neither sent through any message nor stored in the SM_i and SP in the plaintext format. Hence, the proposed scheme is securely infeasible to the attacks on user anonymity.

13) Stolen-verifier attack: If the attacker somehow discovers a smart card/smart device, server SP, he/she could launch a power-analysis attack to know the secret information stored inside. However, in the proposed scheme, during the registration phase, the TS stores $t' = T \oplus CID_i \oplus s \oplus ID_{SP}$, $a' = A_i \oplus CID_i \oplus s \oplus ID_{SP}$ and $e_t' = E_t \oplus CID_i \oplus s \oplus ID_{SP}$ against CID_i. Even if the attacker somehow steals those records, he/she cannot perform the malicious activity because he/she would be unable to access the plain text (T, A, E_t), as the record is protected using the secret key (s) and ID_{SP} of SP and CID_i. The security of s and ID_{SP} in $E_{ks} = s.\ Id_{SP}.G$ is well protected by the concept of ECDLP and CDHP. In addition, he/she cannot create a substantial login solicitation to pass the verification stage without knowing CK', as it is not stored in the server's database. Furthermore, cookie computation, i.e., $CK' = CK \times G$ depends on the correct computation of $CK = h(R||s||E_t||CID_i)$. Without knowing the server's secret key (s), the attacker cannot compute a substantial cookie (CK) and, hence, cannot make a legitimate login request. In addition, it is impossible for an attacker to extract information stored in SM_i due to its tamper-proof design [29]. Therefore, the proposed scheme can withstand stolen-verifier attacks.

14) Cookie-theft attacks: In the proposed scheme, the session cookie (C_k) is stored and sent in the form of $CK' = h(R||s||E_t||CID_i) \times G$, (i.e., an ECC point multiplication) in the embedded device (SM_i). Therefore, it is very difficult to extract CK from CK' because of the complexity of ECDLP. In addition, because of tamper-proof design SM_i, it is impossible for an attacker to extract CK' from it [29]. In addition, CK' is sent through a secure, trusted channel during the registration step which we explain in registration Subsection IV.*A*. Consequently, the attacker cannot get the cookie (CK'). Therefore, the proposed scheme is secure to cookie-theft attacks.

15) Man-in-the-middle attacks: Mutual authentication prevents man-in-the-middle attacks. We verified the mutual authentication of the proposed scheme using ProVerif tool in Subsection IV.*D* [14]. Thus, the proposed scheme successfully supports mutual authentication between SM_i and SP. Consequently, the proposed scheme is secure to man-in-the-middle attacks.

16) Brute-force attacks: To launch a brute-force attack, an attacker must extract the security parameters X_i, Y_i, X_j, Y_j, and T from the transmitted messages. However, even if the attacker succeeds in extracting the parameters, he/she cannot determine the password (psw_i), and also the server's secret key (s) which is obscure and its security protected by the concept of ECDLP and CDHP which we mentioned earlier; furthermore, there is no chance to speculate the random numbers $(x_{i1}$ and $x_{j1})$ because of protection offered by ECDLP and CDHP (i.e., already explain in known session-specific temporary information attacks). Therefore, the proposed scheme can resist brute-force attacks.

Finally, due to the page limit, we selected some security attributes (i.e., usually protect the security of an authentication protocol) for comparison. We compare those security attributes of some current schemes [16-18, 20-25] with those of the proposed scheme, as presented in Table IV. The comparison demonstrates that the proposed scheme is free from all the shortcomings within the existing schemes.

REFERENCES

[34] Darwish, M., Ouda, A. and Capretz, L.F., 2017. Formal Analysis of an Authentication Protocol against External Cloud-Based Denial-of-Service (DoS) Attack. arXiv preprint arXiv:1711.09985.

[35] Abbasinezhad-Mood, D., Ostad-Sharif, A. and Nikooghadam, M., 2020. Efficient provably-secure privacy-preserving signature-based key establishment protocol. Ad Hoc Networks, 100, p.102062.

[33] wolfSSL. "WolfSSL Embedded SSL/TLS Library: Now Supporting TLS 1.3." WolfSSL, www.wolfssl.com/.

[29] Kumari, S., Karuppiah, M., Das, A.K., Li, X., Wu, F. and Kumar, N., 2018. A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers. The Journal of Supercomputing, 74(12), pp.6428-6453.

[14] Maitra, T., Obaidat, M.S., Amin, R., Islam, S.H., Chaudhry, S.A. and Giri, D., 2017. A robust ElGamal-based password-authentication protocol using smart card for client-server communication. International Journal of Communication Systems, 30(11), p.e3242.

[16] Zhou, L., Li, X., Yeh, K.H., Su, C. and Chiu, W., 2019. Lightweight IoT-based authentication scheme in cloud computing circumstance. *Future Generation Computer Systems*, *91*, pp.244-251.

[17] Yu, S., Park, K. and Park, Y., 2019. A Secure Lightweight Three-Factor Authentication Scheme for IoT in Cloud Computing Environment. *Sensors*, *19*(16), p.3598.

[18] Xie, Z. and Jiang, L., 2020, January. An improved authentication scheme for the Internet of things. In IOP Conference Series: Materials Science and Engineering (Vol. 715, No. 1, p. 012031). IOP Publishing.

[20] Chatterjee, S. and Samaddar, S.G., 2020. A robust lightweight ECC-based three-way authentication scheme for IoT in the cloud. In Smart Computing Paradigms: New Progresses and Challenges (pp. 101-111). Springer, Singapore.

[21]Yu, Y., Hu, L. and Chu, J., 2020. A Secure Authentication and Key Agreement Scheme for IoT-Based Cloud Computing Environment. Symmetry, 12(1), p.150.

[22] Sengupta, S., 2020. A Secured Biometric-Based Authentication Scheme in IoT-Based Patient Monitoring System. In Emerging Technology in Modelling and Graphics (pp. 501-518). Springer, Singapore.

[23] Yang, S.K., Shiue, Y.M., Su, Z.Y., Liu, I.H. and Liu, C.G., 2020. An Authentication Information Exchange Scheme in WSN for IoT Applications. IEEE Access, 8, pp.9728-9738.

[24] Wang, F., Xu, G., Xu, G., Wang, Y. and Peng, J., 2020. A Robust IoT-Based Three-Factor Authentication Scheme for Cloud Computing Resistant to Session Key Exposure. Wireless Communications and Mobile Computing, 2020.

[25] Wazid, M., Das, A.K., Bhat, V. and Vasilakos, A.V., 2020. LAM-CIoT: Lightweight authentication mechanism in a cloud-based IoT environment. Journal of Network and Computer Applications, 150, p.102496.