Research Article

Privacy Preserved Data Sharing With SpinalNet Based Key Generation in Consortium Blockchain

Web Intelligence
2025, Vol. 23(3) 395–415

DOI: 10.1177/24056456251325288
journals.sagepub.com/home/web

Suganthi Sekar[1] and T Sree Kala[2]

Abstract

Electronic government (e-government) is the application of communication and information technologies aimed at improving the efficiency of public services provided to citizens and organizations. The e-government is a most complicated system that requires to be distributed, protected as well as privacy-preserved and malfunction of these systems is expensive both socially and economically. Therefore, blockchain technology facilitates the execution of high protected as well as privacy-preserving approaches wherein transactions are not underneath a third party control managements. Utilizing blockchain technology, current data and fresh data are saved in the sealed compartment of blocks that is distributed across a network in demonstrable manner. The information privacy and security are improved by blockchain technology, wherein data are distributed and encrypted across whole network. Here, SpinalNet_KeyGen is newly presented for privacy-preserved data sharing (PPDS) in e-governance system. The entities considered in this research are citizen, private agency, government agency (GA) and support GA. Initialization, delegates and witness voting, key generation wherein secret key is generated using SpinalNet, new node creation, user registration, data encryption and protection, authentication, decryption, validation and data sharing are the steps followed in this work. The devised scheme follows three groups of communication. Moreover, SpinalNet_KeyGen achieved minimal values of computational time and memory usage about 37.526 s and 37.9 MB. However, it is difficult to consider all security parameters in the proposed method.

Keywords

blockchain technology, SpinalNet, key generation, data sharing, e-governance

Received: 25 February 2023; accepted: 31 December 2025

1 Introduction

An incorporation of information technology (IT) into the business offers public services accessible online and increases government effectiveness, thus integration is known as e-governance (Assiri et al., 2020). The e-governance is a term that has been spreading nowadays, but it has increased traction in present years, wherein like transformative and digital government, it is frequently utilized as replacement or substitution for "electronic government (e-government)" term (Bannister & Connolly, 2012; Ranjith Kumar & Bhalaji, 2021). The privacy and security is important in various application fields, like medical systems, energy, transportation, etc. (Khowaja et al., 2023; Li et al., 2022; Srinivas & Mohan, 2022). The e-government method collect, stores and processes an important quantity of secret information about employees, products, citizens, researches, customers, financial grade among others, utilizing electronic computers. Even though, e-governance is promising, there are still security and privacy challenges that should be considered when developing these systems (Wang et al., 2023; Yang et al., 2023). The blockchain technology is better choice for securing e-governance (Assiri et al., 2020). Blockchain has aroused as satisfactory interpretation to administer guarded regionalized system for

[1]Department of Computer Science, Sree Muthukumaraswamy College, Chennai, India
[2]Department of Computer Science, Vels Institute of Science, Technology & Advanced Studies, Chennai, India

Corresponding Author:
Suganthi Sekar, Department of Computer Science, Sree Muthukumaraswamy College, Kodungaiyur, Chennai, India.
Email: dinesh.suganthi@gmail.com

the transferring of data (Mukhopadhyay et al., 2016; Nakamoto, 2008; Ranjith Kumar & Bhalaji, 2021). Blockchain comprises of distributed, sequential chain of blocks and continually rising as finished blocks are added to it with newer group of records (Mukhopadhyay et al., 2016).

In blockchain technology, the blocks contain details and transactions from prior blocks. The distinct linear path from an initial block forever posted to present block exists since all blocks include hash of prior block (Mukhopadhyay et al., 2016). It is immutably distributed ledger that enables safe transactions amongst distributed entities (Boualouache et al., 2021; Dorri et al., 2017). Blockchain has been revealed a count of most important features that include immutability, privacy as well as security and thus, it is a helpful technology for addressing the challenges related to it (Dorri et al., 2017). Public and consortium blockchains are the two kinds of structures for blockchain. In a public blockchain, all entities can construct as well as verify the blocks whereas in consortium blockchain only a set of authorized members are able to perform (Boualouache et al., 2021; Dai et al., 2020). The hybrid blockchain is an incorporation of public as well as consortium or private blockchains (Jain & Jalia, 2021). In consortium or permissioned blockchain, count of nodes taking part in consensus is very few and hence this kind of blockchain is able to attain very rapid consensus (Dai et al., 2020; Malik et al., 2022).

A ledger offers the auditing trail of supply chain occurrence, hence paying traceability and provenance of products. As the events are connected with digital identifier of supply chain contributors, every peer accessing a ledger has entire details about product, its trading flows or participants associated to activities of supply chain (Malik et al., 2022). The authentication is a process to determine something or someone it is declaring is a vital element of any dependable online system that controls transactions or sensitive data (Lim et al., 2018). The data of government generally involve private information (Shailaja & Guru Rao, 2019) of enterprises or individuals. The current research utilizes privacy protection technology (Mandala & Chandra Sekhara Rao, 2019) to learn the leakages of sensitive details, which may present on government data. The anonymity-enabled techniques (Sei et al., 2017; Shyamala Susan & Christopher, 2016; Sweeney, 2002), encryption-enabled techniques (Gomez Marmol et al., 2012; Lu et al., 2012; Omori & Yamashita, 2020; Shen et al., 2020; Vu et al., 2020), and differential privacy-enabled techniques (Dwork July 10–14, 2006, Proceedings, Part II 33; Dwork et al. March 4–7, 2006, Proceedings 3) are the three general kinds of privacy security techniques. The e-governance is a method that intends to improve government's capability for simplifying every process, which involves businesses, government, citizens, and so forth. The quick development of digitalized technologies has frequently created a need for establishment of e-governance method. If the data of government is shared without accurate safety measure, private details are leaked easily. In addition, an encryption function brings several computational overhead that makes the encryption complicated for applying resource-constrained scenarios (Piao et al., 2021). Moreover, anonymity-enabled techniques generally lack severe privacy protection guarantees for larger data. These challenges motivated to design a new technique for PPDS in e-governance system.

The basic purpose of this work is to design PPDS approach with SpinalNet_KeyGen in e-governance system. The data protection stage is using data transformation, encryption, XOR operation, hashing function and interpolation. In key generation step, the secret key is generated utilizing SpinalNet. The three groups of communication devised scheme follows in namely government to business (G2B), government to government (G2G) and government to citizen (G2C).

Contribution of this research:

- **SpinalNet_KeyGen for PPDS in e-governance system:** For enhancing the privacy the data protection stage is using data transformation, encryption, XOR operation, hashing function and interpolation. Also, in key generation step, the secret key is generated utilizing SpinalNet.

The beneath sections are organized as: The existing schemes reviewed are interpreted in Section 2, SpinalNet_KeyGen technique is explicated in Section 3 and results achieved by designed approach are elucidated in Section 4 and Section 5 elucidates conclusion of SpinalNet_KeyGen.

2 Literature Survey

Ranjith Kumar and Bhalaji (2021) developed Chameleon Hashing method for privacy preservation in the e-governance. It has the capability for assuring data authentication and confidentiality as well as it increases confidence of public zones. This method failed to consider security against damaging outcomes of the quantum computing in blockchain field. Wang et al. (2019) devised electronic health records (EHR) sharing for privacy preservation through consortium blockchain obtained highest computational effectiveness, but the consumption of gas was increased. Boualouache et al. (2021) introduced utility-based delegated byzantine fault tolerance (U-DBFT) for privacy preservation in 5G-based vehicular fog computing provided rapid as well as consistent consensus procedures, even though privacy level was decreased. Malik

et al. (2022) presented PrivChain for protecting sensible data on the blockchain utilizing zero knowledge evidences. It offered traceability and provenance without showing sensible details to the end-consumers or else entities of supply chain.

Assiri et al. (2020) represented the hierarchical method that involves the utilization of blockchain among de-militarized zone (DMZ) as well as Secured Intranet zone. It was facilitated finest and securable management of significant operations within an organization. This method failed to leverage on the blockchain technology for securing e-governance system in past. Piao et al. (2021) devised local differential privacy (LDP)-based method for protecting private details when sharing the statistics between government organizations. It efficiently reduced statistical mistakes and improved the usefulness of data afterwards privacy security with several distributions as well as sizes of data domain. This technique did not consider ensuring usefulness and sensitivity of the data in case of multi-value sensitive characteristics. Chen et al. (2020) introduced GovChain for facilitating seamless data sharing across the government organizations. This scheme was highly efficient and feasible, even though it only concerned a performance of the smart contracts that are running on blockchain. Elisa et al. (2018) designed blockchain-enabled e-government framework for ensuring information privacy and security, but failed to explore the whole potent in real-time environment.

Wang et al. (2021) implemented a BlockSLAP method for cryptography-based authentication to the smart grid. Here, cutting-edge based blockchain approach was used to offer the security. Also, it removed the common attacks in the practical environment. However, dynamic and batch verification issues were possible. Zhang et al. (2021) established data sharing and storage approach for blockchain-enabled mobile-edge computing. Here, a unique signature private key was utilized for the edge node for homomorphic encryption and data signature. This approach avoided data floods and enhanced the fault tolerance. However, cost of this model was high. Guo et al. (2024) established an approach, named federated and encrypted data store via consortium blockchains (FedEDB) for supporting the privacy-preserving and multi-owner queries. Here, for the security purpose a practical key aggregation scheme was constructed. This scheme offered reliable and secured outcomes with fairness. However, the number of blockchain nodes or data owners increased, the system faced performance bottlenecks due to the increased query complexity and the verification process. Uma Maheswari et al. (2024) established an average hybrid leader optimization (AHLO)-PrivPresKey Gen scheme for secure data sharing based on blockchain. Various safety operators, such as one time password token, passwords, segmenting interpolation, and 3 Data Encryption Standard were used by the AHLO-PrivPresKey Gen scheme. Also, the privacy of the model was improved by different functionalities, like keys, Chebyshev polynomial, and dyadic product and the AHLO was utilized for the key generation. It reduced the user's clarification burden during the information-sharing process, but the computational cost associated with encryption and smart contract execution was high.

2.1 Challenges

The methods above reviewed faced some demerits in case of privacy preservation in e-governance systems that are elucidated as follows.

- Chameleon Hashing method in Ranjith Kumar and Bhalaji (2021) was designed for privacy preservation in a system of e-governance did not consider the development of blockchain modernism like Ethereum in public region for improving security and privacy of data.
- In Wang et al. (2019), HER sharing was developed for privacy preservation through consortium blockchain attained devised security goals, but still it failed to implement the model on Hyper-ledger Fabric as well as perfect smart contracts to run the approaches of the data sharing.
- The technique termed U-DBFT designed in Boualouache et al. (2021) for privacy preservation in 5G-based vehicular fog computing was failed to examine adaptive methods that consider extra parameters like traffic density and mobility in optimized smart contract (OSC) creation.
- PrivChain introduced in Malik et al. (2022) acquired minimum overheads for evidence verification as well as encryption related chores, though it did not investigate other variants of the zero knowledge evidences like Bulletproofs.

3 Proposed Spinalnet_Keygen for PPDS in E-Governance System

Here, SpinalNet_KeyGen is presented for PPDS in an e-governance system. In this work, citizen, support government agency (GA), private agency and GA. This model consists of four layers such as consortium blockchain, service access, ledger storage and network layer. The service access layer consists of e-government user details and diverse devices to provide computing resources, accessing and credential data storages. In consortium blockchain layer, participants are pre-chosen government organizations that are responsible to validate transactions as well as authenticate e-government

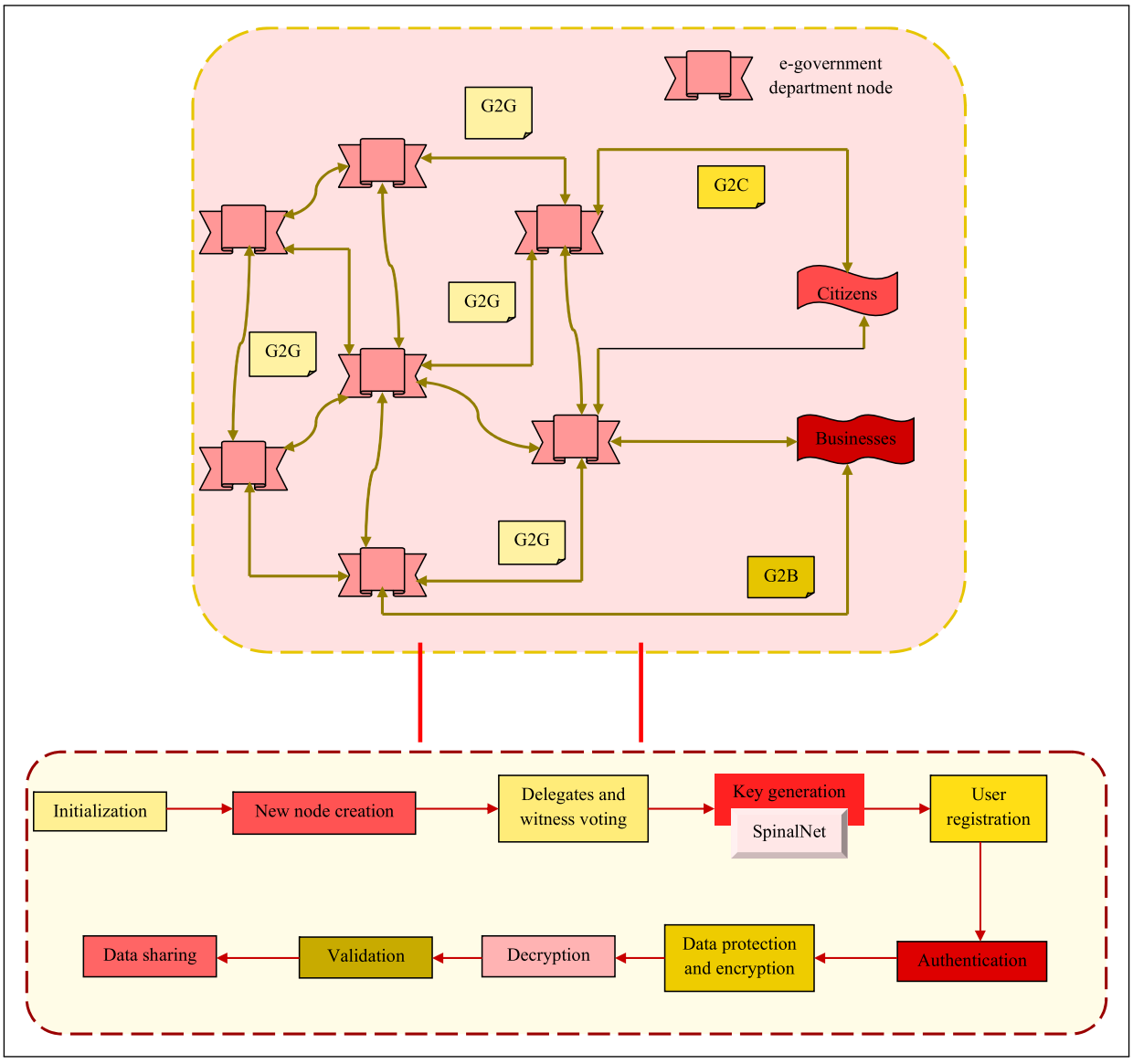

Figure 1. Pictorial view of SpinalNet_KeyGen.

user registration with a consortium blockchain network. A network layer provides connectivity among ledger storage servers, consortium blockchain layer, government and e-government users. A ledger storage layer is provided for enabling storage as well as duplication of users and the government details. The steps followed are initialization, delegates and witness voting, user registration, new node creation, key generation, authentication, validation, data protection and data sharing. In key generation step, SpinalNet_KeyGen is utilized for secret key generation. The data protection is on basis of data transformation, XOR operation, encryption, decryption, hashing function and interpolation. The devised technique follows communication in G2C, G2G and G2B groups. The G2C exchanges data amongst people and legislature, G2G specifies the communication happening among government sections as well as associations whereas G2B refers to data trade communications. Figure 1 demonstrates pictorial view of devised SpinalNet_KeyGen for PPDS in e-governance system.

Table 1 reveals the description of symbols utilized this article.

3.1 Set of Phases

The phases involved are initialization, delegates and witness voting, new node forming, key generation, user registration, data protection, authentication, encryption, decryption, validation and data sharing.

Table 1. Description of Symbols.

Symbol	Description
y, z, γ	Random number
$\hbar(.)$	Hashed function
N_I	ID of new node
N_W	Password of new node
N_ω	New node
N_K	Private key of new node
B_p	Public key of newer node
SK	Secret key
$\mathfrak{R}$	Record
G_I	ID of support agency
G_W	Password of support agency
E_1, E_2, E_3, E_4	Verification messages
V_I	ID of Private agent
V_W	Password of private agent
Q	Message
U_1, U_2, U_3	Authentication messages
M	Timestamp
Z_I	ID of citizen
Z_W	Password of citizen
$T(.)$	Encryption function
Y_J	Yeo-Johnson data transformation
k	Key
c	Chebyshev polynomial
H_W, H_{W2}	Session passwords
$\mathfrak{R}_{q1}, \mathfrak{R}_{q2}$	Records

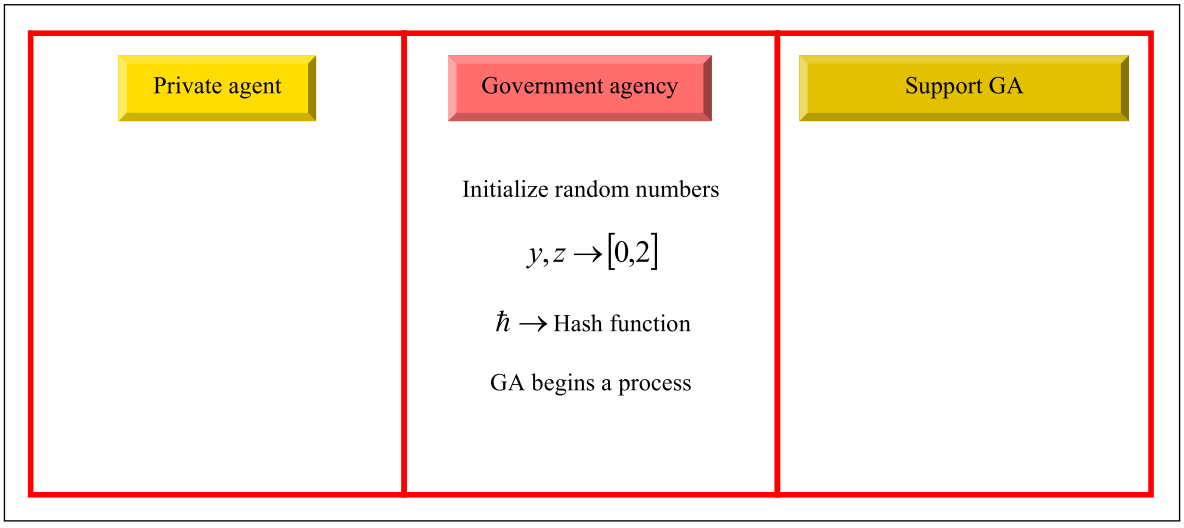

Figure 2. Initialization phase of SpinalNet_KeyGen.

3.1.1 Initialization. At initialization phase, random numbers are firstly initialized. The initialized random numbers are denoted by y and z that are fixed to lie among 0 and 2. Thereafter, hash function $\hbar$ is initialized and then, GA begins a process. The initialization of SpinalNet_KeyGen is shown in Figure 2.

3.1.2 New Node Creation. A registration procedure for e-governance network is elucidated beneath. Firstly, new node N_ω is generated by concatenating newer node password N_W and private key N_K, in which hashed function is then performed. The hashed outcome is thereafter performs XOR operation with y. Then, the obtained outcome is multiplied with ID of

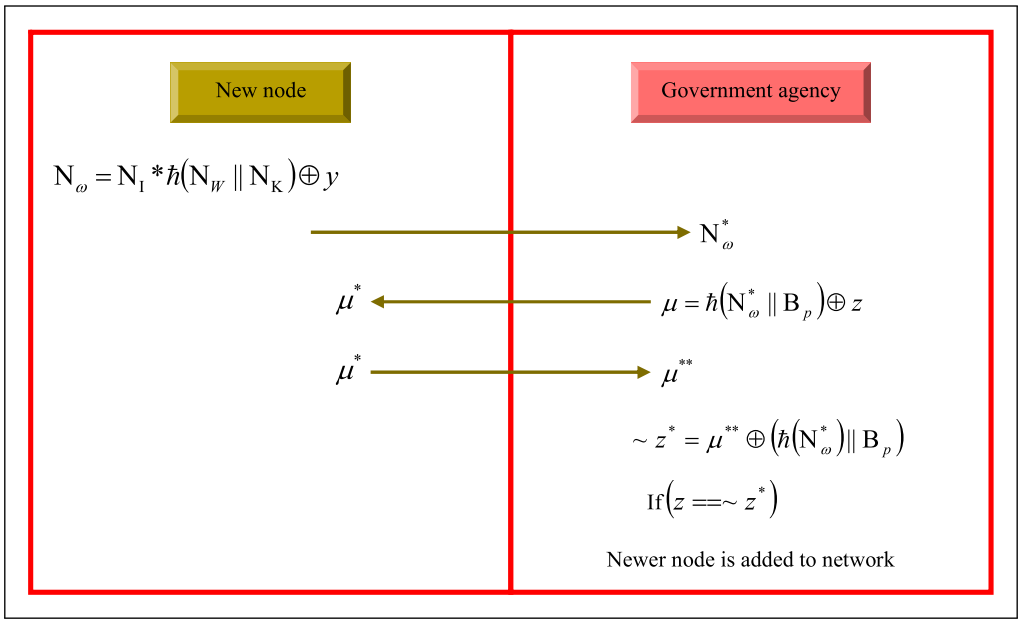

Figure 3. New node creation of SpinalNet_KeyGen.

new node N_I, which can be expressed by,

$$N_\omega = N_I * \hbar(N_W || N_K) \oplus y \tag{1}$$

Afterwards, new node thus obtained is stored by the GA as N_ω^* and thereafter, a message is created by GA. The message is generated by concatenation of newer node N_ω^* with new node public key B_p and thereafter the result is hashed. Then, hashed outcome performs XOR operation with random number z. It can be represented as follows,

$$\mu = \hbar(N_\omega^* || B_p) \oplus z \tag{2}$$

The message created by GA is passed to new node that is stored as μ^*, which is then again passed to GA and it is stored as μ^{**}. Then a result performs XOR operation with the stored message μ^{**} and can be modeled by,

$$\sim z^* = \mu^{**} \oplus (\hbar(N_\omega^*) || B_p) \tag{3}$$

After that, equality $(z = \sim z^*)$ is evaluated, and the condition is satisfied, then newer node creation is done, Figure 3 demonstrates newer node creation of SpinalNet_KeyGen.

3.1.3 Delegates and Witness Voting. In this phase, new node N_ω is generated and it is stored as N_ω^* by private agent. A private agent is in charge of a voting and while newer node is added, private agent authenticates newer node. At first, newer node generates password N_W and ID denoted by N_I. It is then given to private agent and a condition is applied. If $(N_I = N_I^*)$ and $(N_W = N_W^*)$, then a newer node is authenticated and if more than one newer node is added for selecting newer citizen, then the voting is performed. Then, GA generates secret key SK. The hashed ID of citizen $\hbar(Z_I^*)$ as well as hashed password of citizen $\hbar(Z_W^*)$ are concatenated and additionally modulus of random number γ is applied. Hence, an attained outcome is performed for generating secret key that can be formulated by,

$$SK = (\hbar(Z_I^*) || \hbar(Z_W^*)) \bmod \gamma \tag{4}$$

Thereafter, secret key is saved by newer node as SK^* and by private agent as SK^{**}. Afterwards registration, citizen submit a record $\mathfrak{R}$ to GA to store record in the blockchain. These records are stored by GA as $\mathfrak{R}^*$, where γ implies random number. Delegates and witness voting of SpinalNet_KeyGen is revealed in Figure 4.

3.1.4 Key Generation Utilizing SpinalNet_KeyGen. Here, generation of secret key is carried out utilizing SpinalNet. The below subsection describes the architecture of SpinalNet for secret key generation.

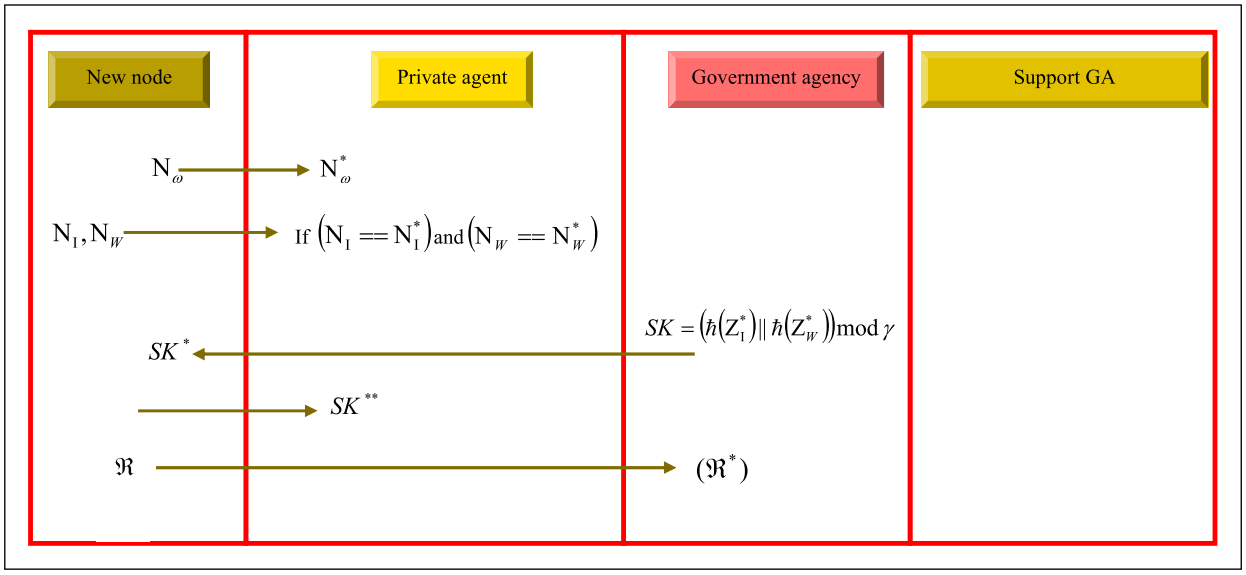

Figure 4. Delegates and witness voting phase of SpinalNet_KeyGen.

3.1.4.1 Architecture of SpinalNet. The developed SpinalNet (Dipu Kabir et al., 2022) has similarities shown below in accordance to features of human spinal cord.

- Gradual input
- Local output as well as feasible global influence
- Reconfigured weights at the time of training

Same like human spinal cord, the SpinalNet acquires input in gradual manner and repeatedly. Individual layer of SpinalNet contributes against local outputs. SpinalNet also transfers transformed version of the inputs against global outputs. The neural network (NN) training procedure organizes weights on basis of training data. The neurons of spinal cord are also gets configured to tune a pain sensitivity of diverse sensors of human body.

The structure of SpinalNet includes input sub-layers, intermediary sub-layers, and an output layer. Inputs are segmented and directed to intermediary sub-layers with several hidden layers, each having two neurons. Users can modify the number of neurons in these sub-layers, and typically, fewer neurons and smaller input sizes are used to decrease the multiplication count. In general, count of inputs as well as intermediary hidden neurons for each layer allocates a small quantity, and then a network may have the under-fitted shape. As an outcome, an individual layer acquires inputs from prior layer. Thus an input is repetitive, if the significant input feature does not impact an output in single hidden layer; a feature may impact an output in other hidden layers. An intermediary sub-layers consists of non-linear activation function whereas an output layer contains linear activation function.

From this process, secret key is obtained from an inner layer of SpinalNet and Figure 5 elucidates architecture of SpinalNet.

3.1.5 User Registration Phase. The users register by means of gadgets or else manually visit any one of departments. At first, citizen is registered with the GA by creating ID of citizen Z_I as well as password Z_W. It is then given to GA and is stored as Z^*_I and Z^*_W and then, private agent saves it as Z^{**}_I and Z^{**}_W. After that, verification message is produced by GA, which is generated by concatenating hashed citizen password with y and then, it performs XOR function with z. It can be formulated by,

$$E_1 = (\hbar(Z^*_W)||y) \oplus z \tag{5}$$

A verification message-1 thus obtained is saved by private agent as E^*_1 and then by GA as E^{**}_1. Thereafter, $\sim z^*$ is evaluated, wherein hashed password of citizen is concatenated along with a random number y and afterwards, XOR function is applied with saved verification message-1 E^{**}_1. It can be represented by,

$$\sim z^* = E^{**}_1 \oplus (\hbar^*(Z^*_W)||y) \tag{6}$$

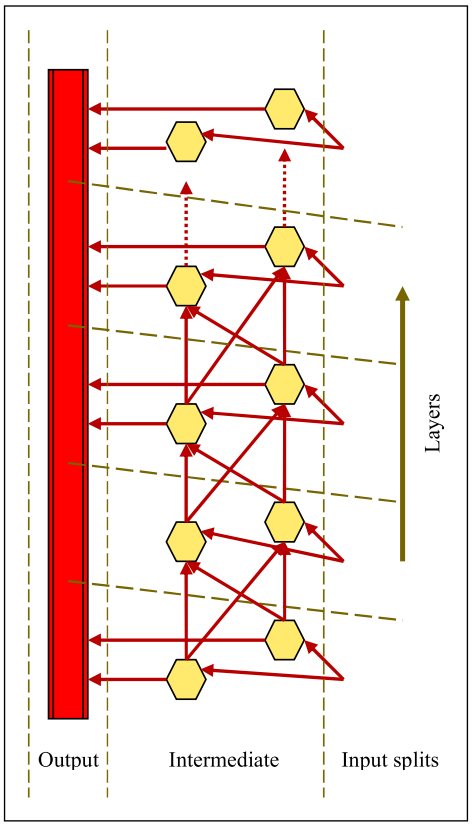

Figure 5. Architecture of SpinalNet.

If $(z = \sim z^*)$, then citizen is registered. A support GA has to be register with the government agency. Hence, support GA ID as well as password is attained, which is indicated as G_{I} and G_W. It is then stored by the GA as G_{I}^* and G_W^*, which is additionally stored by the private agent as G_{I}^{**} and G_W^{**}. Afterwards, verification message-2 is created by means of hashed password of support GA $\hbar(G_W^*)$ and is then concatenated with the random number y, which is then performs XOR function with z and can be illustrated by,

$$E_2 = (\hbar(G_W^*)||y) \oplus z \tag{7}$$

Thereafter, $\sim z^*$ is evaluated, where hashed password of support GA is concatenated with the random number y and afterwards, XOR operation is performed with stored verification message-2 E_2^{**}, which can be given by,

$$\sim z^* = E_2^{**} \oplus (\hbar(G_W)||y) \tag{8}$$

If $(z = \sim z)$, support GA registration takes place and then, private agent ID as well as password specified by V_{I} and V_W is acquired, which is thereafter stored by GA as V_{I}^* and V_W^*. After that, registration message is attained by concatenation of stored private data V_{I}^* and V_W^* and then it is hashed. Then, modulus of random number γ is applied that can be expressed by,

$$Q = \hbar(V_{\mathrm{I}}^*||V_W^*) \bmod \gamma \tag{9}$$

A registration message is then saved by private agent as Q^* and it is saved by the GA as Q^{**}. If $(Q = (\sim Q = \hbar(V_I^*||V_W^*) \bmod \gamma))$ is satisfied, then private agent is verified. Figure 6 shows user registration phase of SpinalNet_KeyGen.

The registration message is stored by private agent as D^* and the GA stores it as D^{**}. If, $(D = (\sim D = h(P_{ID}^*||P_{PW}^*) \bmod r))$, then private agent is verified. Figure 5 displays the user registration phase of designed privacy protected model.

3.1.6 Authentication Phase. An authentication phase is carried out among citizen and the GA. At first, authentication message-1 is created. Here, XOR operation is performed with random number y as well as timestamp M. Then, it is

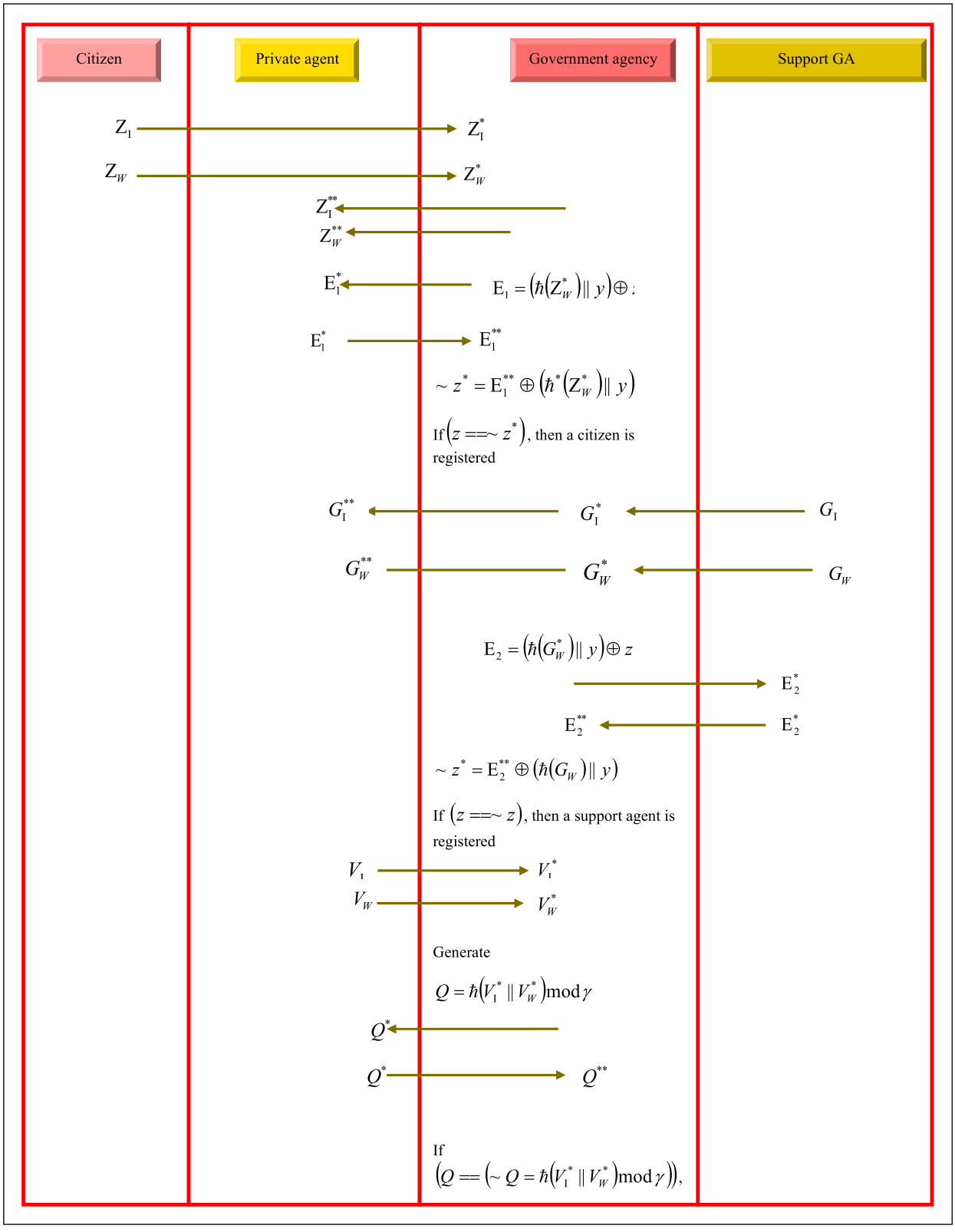

Figure 6. User registration phase of SpinalNet_KeyGen.

concatenated with the hashed secret key and ID of citizen that can be illustrated by,

$$U_1 = \hbar(SK) \,||\, y \oplus M \,||\, \hbar(\mathrm{Z_I}) \tag{10}$$

Here, M signifies timestamp. Thereafter, GA verifies timestamp validity. If it is verified as yes, then a session continues or else it terminates. An authentication message is evaluated by GA by performing XOR operation with random number y and timestamp M.

$$\sim U_1 = \hbar(SK) \,||\, y \oplus M \,||\, \hbar(\mathrm{Z_I}) \tag{11}$$

If $(U_1 = \sim U_1)$, citizen is authenticated to the GA and an authentication is performed among support agency as well as GA. Here, support agency evaluates authentication message-2 by concatenating hashed ID of support agency, hashed password of support agency and the timestamp.

$$U_2 = \hbar(G_\mathrm{I})||\hbar(G_W)||M \tag{12}$$

Then, authentication message-2 is subjected to the GA. A GA verifies timestamp validity. If it is verified as yes, then a session continues or else it terminates. The computation of authentication message-2 is done by concatenating hashed ID of support agency, hashed password of support agency and the timestamp that can be expressed by,

$$\sim U_2 = \hbar(G^*_\mathrm{I})||\hbar(G^*_W)||M \tag{13}$$

If $(U_2 = \sim U_2)$, then a support agency is authenticated to GA and a private agency evaluates an authentication message-3 by concatenating hashed ID of private agent, hashed password of private agent and the timestamp. The expression can be given by,

$$U_3 = \hbar(V_\mathrm{I})||\hbar(V_W)||M \tag{14}$$

Thereafter, authentication message-3 is given to GA and a GA verifies validity of timestamp. If the verification is yes, then the session continues or else it terminates. If $(U_3 = \sim U_3)$, the private agency is authenticated with the GA and authentication of SpinalNet_KeyGen is delineated in Figure 7.

3.1.7 Data Protection and Encryption Phase. The data protection is carried out among private agent and GA. A data that is communicated will be secured by preserving privacy of data utilizing following model.

Let the database be $D_{r\times s}$ and then, Yeo-Johnson data transformation is applied that can be given by,

$$D^Z = Y_J(D) \tag{15}$$

The coded data can be illustrated by,

$$P_{r\times s} = T(D^Z, k) \tag{16}$$

Where, k implies key and can be formulated by,

$$k = \sum_{l=1}^{b} D_l \oplus c \oplus SK \tag{17}$$

In the above equation, SK signifies secret key whereas c implies Chebyshev polynomial and it can be modeled by,

$$c = 30i^2 + 10i + 5 \tag{18}$$

$$i = i_1 + \frac{i_2 - i_1}{j_2 - j_1}(j - j_1) \tag{19}$$

The values of $j = 12$; $i_1, j_1 = 21$ and $j_2, i_2 = 23$. Let us consider,

$$A_{r\times s} = P_{r\times s} \oplus L_{s\times s} \tag{20}$$

Where, L specifies identity matrix and $\oplus$ implies XOR operation.

$$X_{r\times s} = A + R \tag{21}$$

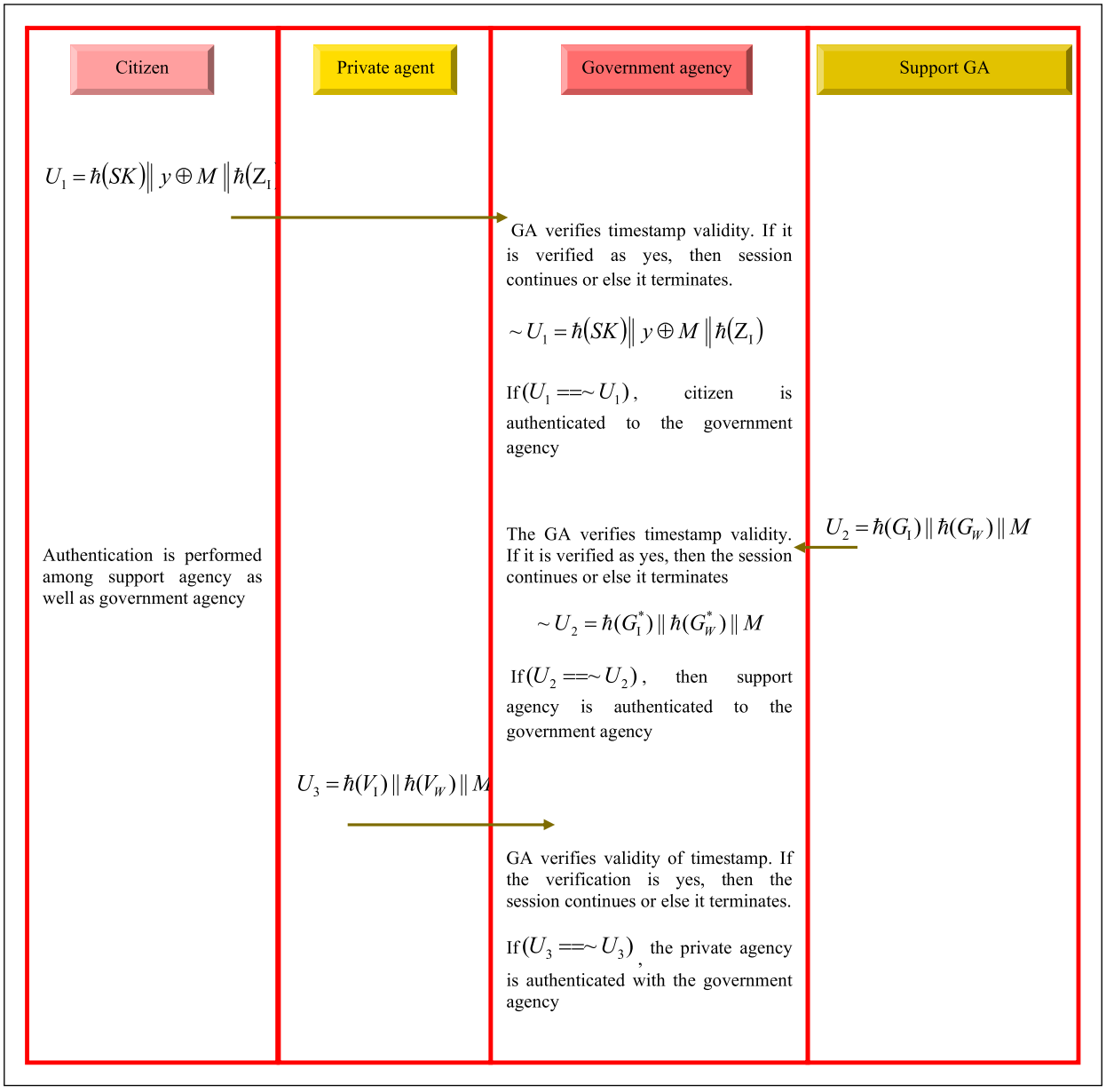

Figure 7. Authentication phase of SpinalNet_KeyGen.

Here, + indicates element-wise addition. The privacy-preserved data can be formulated by,

$$B_h = X_h \bullet D_h^Z \tag{22}$$

In the above equation, • represents matrix multiplication. The privacy-preserved data is then stored by a GA as B_h^*. Thereafter, k is given to the GA and it is stored as k^*. Figure 8 demonstrates data protection and encryption phase of SpinalNet_KeyGen.

3.1.8 Decryption Phase. In decryption phase, data is decrypted in a GA. In this phase, decrypted message is generated by decrypting the saved decrypted data D_t with the saved key k. This can be illustrated by below mentioned equation.

$$A = D_t(B_h^*, k) \tag{23}$$

Here, A symbolizes accessed data. Figure 9 illustrates decryption phase of SpinalNet_KeyGen.

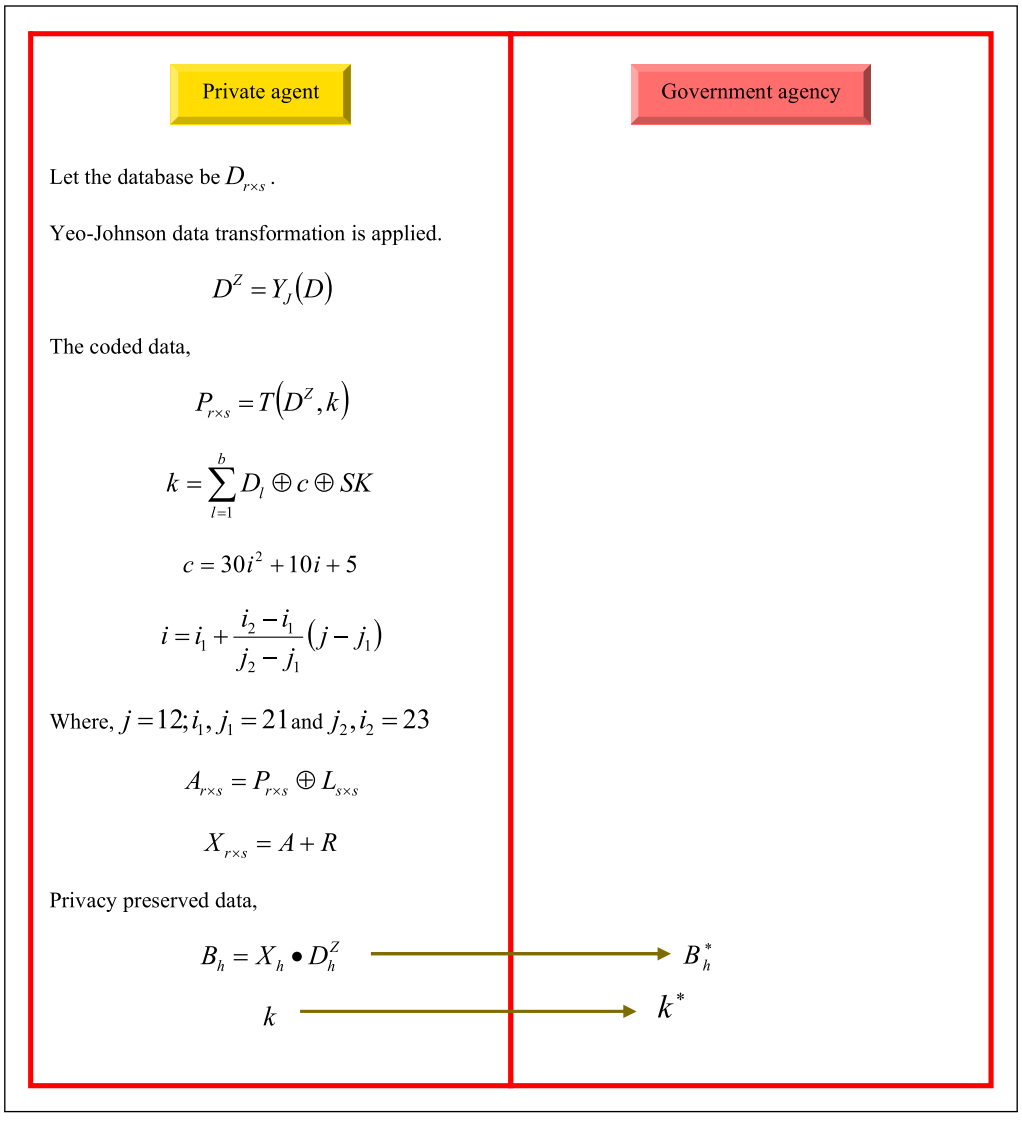

Figure 8. Data protection and encryption of SpinalNet_KeyGen.

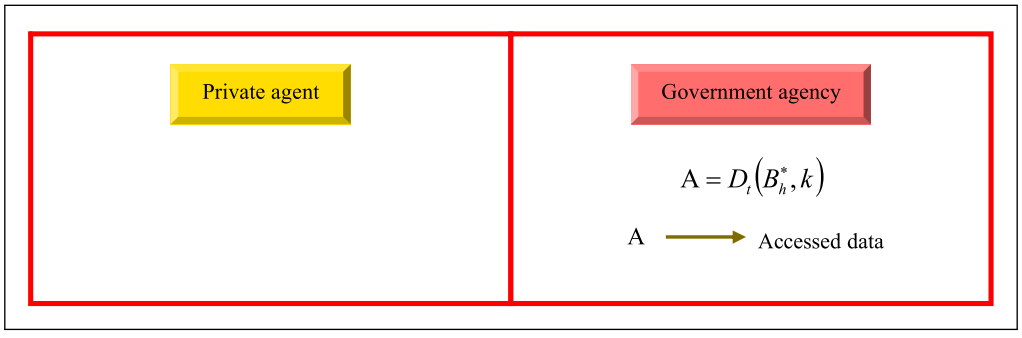

Figure 9. Decryption phase of SpinalNet_KeyGen.

3.1.9 Validation Phase. At first, a private agent requests the data from GA. Then, GA confirms the validation and thereafter provides a record to the private agency. This private agent acquires ID denoted by V_{I} and password V_W, which is then given to GA, wherein session password is produced. It can be illustrated as follows.

$$H_W = \hbar(V_{\mathrm{I}}^{**} + \hbar(SK) + V_W^{**}) \tag{24}$$

Here, H_W indicates session password. Then, session password is stored in the private agent as H_W^*. This can be expressed as shown below.

$$E_3 = \hbar(H_W^* \oplus (V_W)) \tag{25}$$

Similarly, GA generates verification message-3, where stored session password as well as private agent password perform XOR operation and thereafter it is hashed that can be specified by,

$$\sim E_3 = \hbar(H_W \oplus (V_W)) \tag{26}$$

If $(E_3 = \sim E_3)$, then the private agent is validated and GA transfers information of record to a private agency. Thereafter, support GA acquires ID signified as G_I and password denoted by G_W. Then, GA creates session password by summing stored ID of support GA and hashed secret key, which is then hashed to generated session password-2. It is illustrated by a formula mentioned beneath.

$$H_{W2} = \hbar(G_{\mathrm{I}}^* + \hbar(SK)) \tag{27}$$

A session password-2 is then stored by the support GA as H_{W2}^*. After that, support agency generates verification message-4 by performing XOR operation between stored session password-2 as well as password of support agency and then, it is hashed that can be modeled by,

$$E_4 = \hbar(H_{W2}^* \oplus G_W) \tag{28}$$

In the same manner, GA generates verification message-4, where XOR operation is performed with stored session password-2 as well as password of support agency and thereafter, it is hashed. It can be illustrated as follows.

$$\sim E_4 = \hbar(H_{W2} \oplus G_W^*) \tag{29}$$

If $(E_4 = \sim E_4)$, then the support agency is validated and thereafter, GA transfers record to a support agency. The validation phase of SpinalNet_KeyGen is explicated in Figure 10.

3.1.10 Data Sharing Phase. At first, ID of private agent signified by V_{I} and password of private agent denoted by V_W are acquired and passed to the GA. If $(V_{\mathrm{I}} = V_{\mathrm{I}}^*)$ and $(V_W = V_W^*)$, then a process starts. Consider $\mathfrak{R}$ as the record. At this phase, record-1 specified as $\mathfrak{R}_{q1}$ is attained by hashing ID of private agent and then, concatenating with record $\mathfrak{R}$ that can be formulated by,

$$\mathfrak{R}_{q1} = \mathfrak{R} || \hbar(V_{\mathrm{I}}) \tag{30}$$

The private agent stores a record-1 as $\mathfrak{R}_{q1}$. The ID as well as password indicated by G_{I} and G_W is generated by support agency, which is thereafter given to the GA. If $(G_{\mathrm{I}} = G_{\mathrm{I}}^*)$ and $(G_W = G_W^*)$, then a record $\mathfrak{R}_{q2}$ is transferred to the support agency.

Here, record-2 denoted by $\mathfrak{R}_{q2}$ is acquired by concatenating hashed ID of support GA and record $\mathfrak{R}$, which can be modeled by,

$$\mathfrak{R}_{q2} = \mathfrak{R} || \hbar(G_{\mathrm{I}}) \tag{31}$$

Lastly, record-2 is stored in the support GA in a form as $\mathfrak{R}_{q2}^*$. Figure 11 interprets data sharing phase of SpinalNet_KeyGen.

4 Results and Discussion

SpinalNet_KeyGen devised for PPDS in an e-governance system achieved better outcomes by assessing with other approaches. The achievements of SpinalNet_KeyGen are discussed in this section along with experimental setup, description of dataset and performance measures utilized.

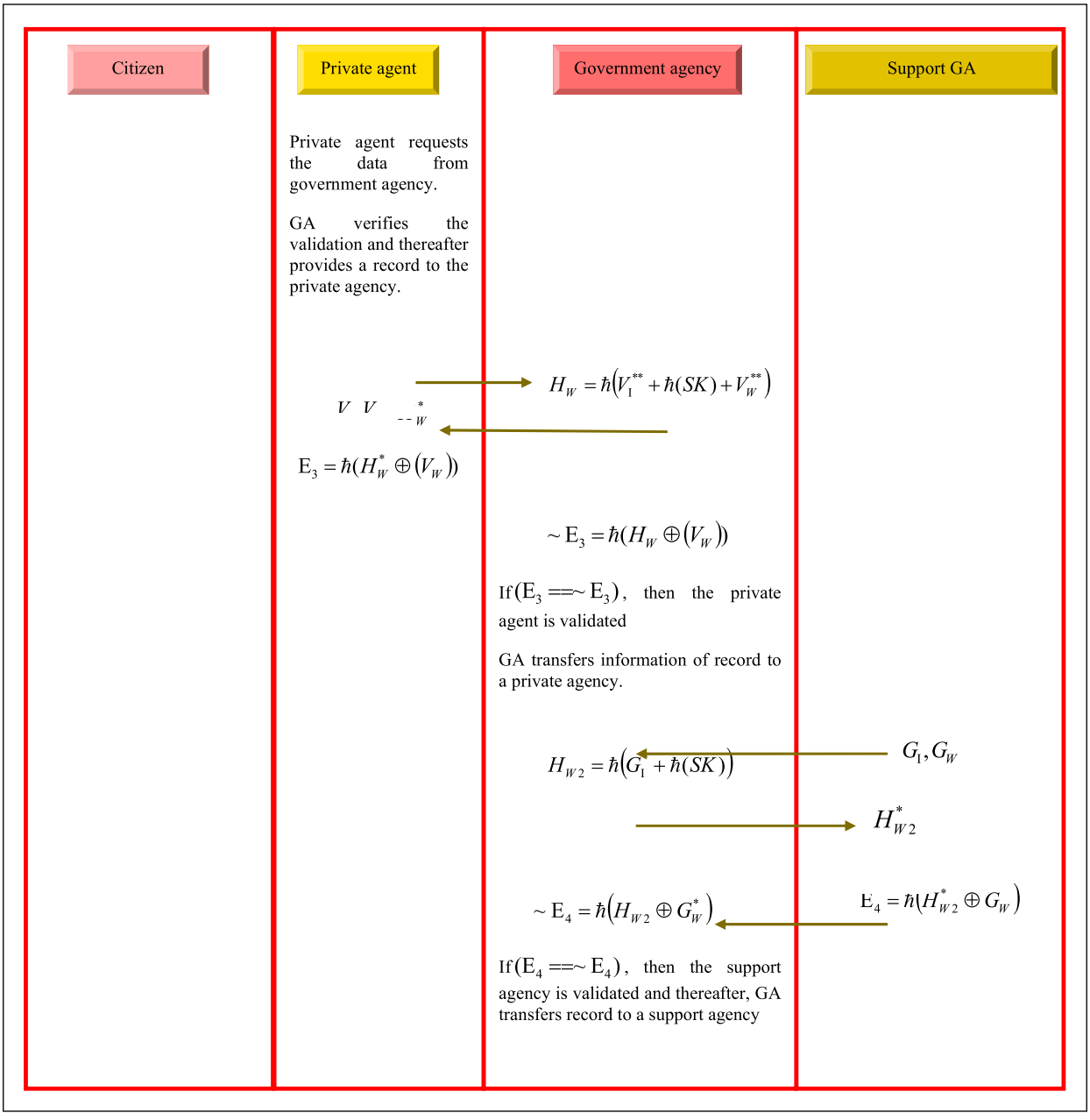

Figure 10. Validation phase of SpinalNet_KeyGen.

4.1 Experimentation Setup

SpinalNet_KeyGen devised for PPDS in e-governance system is implemented in PYTHON tool. Table 2 represents the experimental parameters of the SpinalNet_KeyGen.

4.2 Description of Database

The dataset used in this research comprises of cases happened in various states listed by district-wise from year 2001 to 2014. The cases include kidnapping and abduction, rape, dowry deaths, assault on women with intent to outrage her modesty and insult to modesty of women.

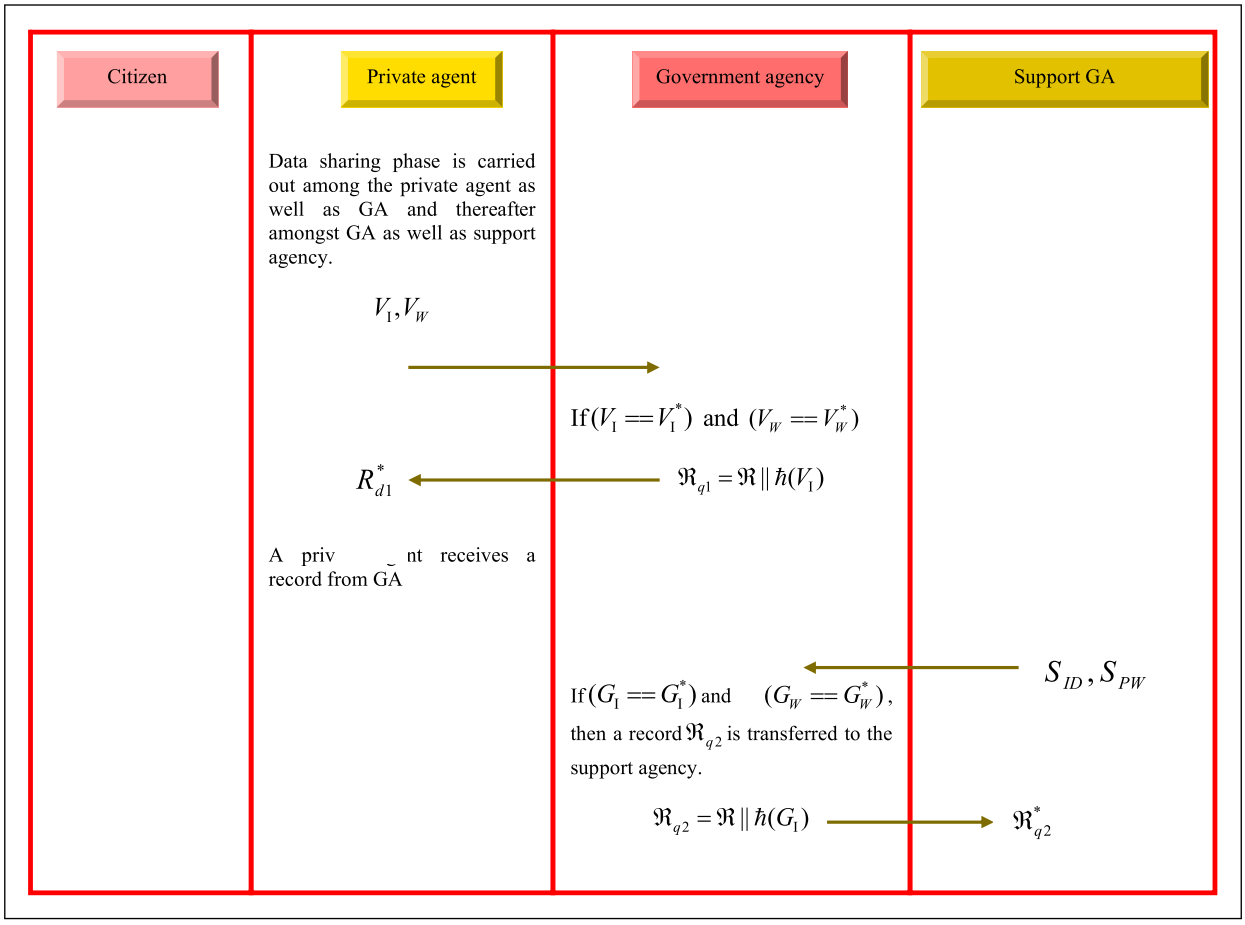

Figure 11. Data sharing phase of SpinalNet_KeyGen.

Table 2. Experimental Parameters.

Parameter	Value
Batch size	50
Input shape	(32, 32, 3)
Epochs	20
Learning rate	0.01

4.3 Performance Metrics

The measures for evaluation utilized to assess SpinalNet_KeyGen are computational time and memory usage that are interpreted in below sub-sections.

4.3.1 Computational Time. The computational time is referred as a time length needed for performing computation process. This metric determines whether an approach is capable to run within particular time frame.

4.3.2 Memory Usage. The percentage of memory that is utilized during a sample period is known as memory usage. This metric finds the memory, which is required to run an application on the instance.

4.4 Performance Analysis

A performance evaluation of SpinalNet_KeyGen is performed by varying key length with several citizens based upon 1 GB, 1.5 GB and 2 GB considering performance measures.

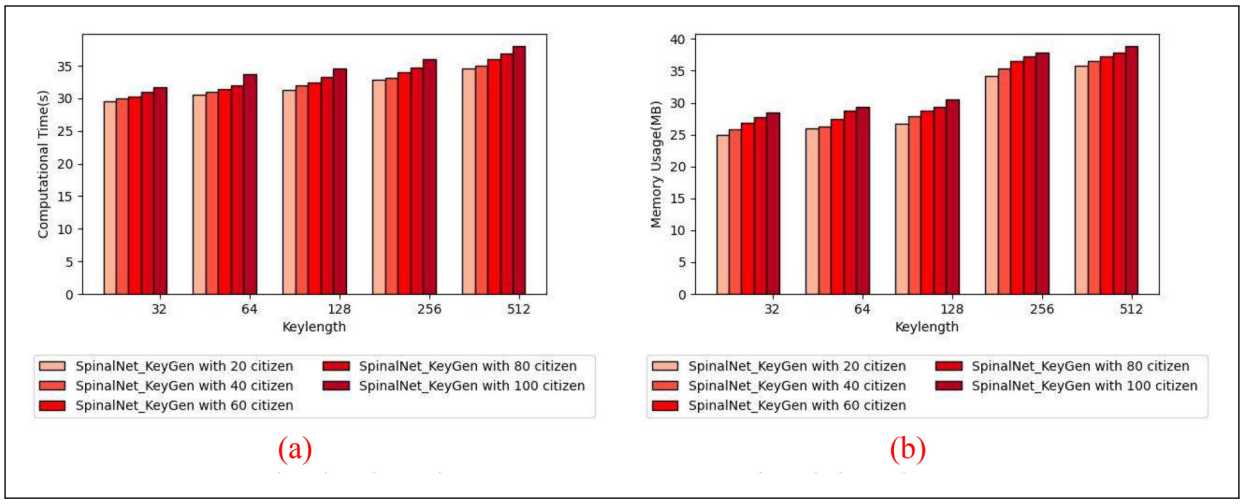

Figure 12. Evaluation based upon 1 GB: (a) computational time and (b) memory usage.

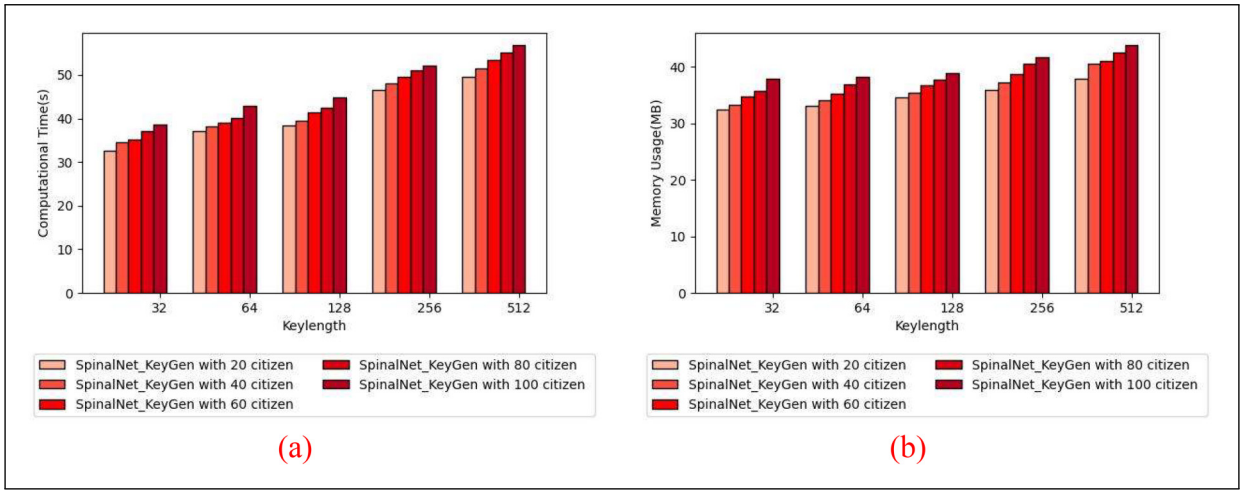

Figure 13. Assessment based upon 1.5 GB: (a) computational time and (b) memory usage.

4.4.1 Assessment Based Upon Data Size = 1 GB. SpinalNet_KeyGen is estimated for revealing a performance by changing key length with several citizens is explicated in Figure 12. Figure 12(a) delineates performance analysis of SpinalNet_KeyGen in terms of computational time. For key length = 512, computational time attained by SpinalNet_KeyGen is 34.526 s, 35.025 s, 35.977 s, 36.847 s and 38.049 s with 20 citizen, 40 citizen, 60 citizen, 80 citizen and 100 citizen. The estimation of SpinalNet_KeyGen based upon memory usage is illustrated in Figure 12(b). Memory usage obtained by SpinalNet_KeyGen is 35.8 MB with 20 citizen, 36.5 MB with 40 citizen, 37.2 MB with 60 citizen, 37.9 MB with 80 citizen and 38.9 MB with 100 citizen while key length is 512.

4.4.2 Assessment Based Upon Data Size = 1.5 GB. Figure 13 illustrates assessment of the SpinalNet_KeyGen by varying key length with several citizens. Evaluation of SpinalNet_KeyGen regarding computational time is elucidated in Figure 13(a). Computational time achieved by SpinalNet_KeyGen is 49.573 s with 20 citizen, 51.427 s with 40 citizen, 53.426 s with 60 citizen, 55.125 s with 80 citizen and 56.892 s with 100 citizen for key length is 512. Figure 13(b) shows analysis of SpinalNet_KeyGen in terms of memory usage. When key length = 512, memory usage acquired by SpinalNet_KeyGen is 37.9 MB, 40.5 MB, 41.1 MB, 42.6 MB and 43.9 MB with 20 citizen, 40 citizen, 60 citizen, 80 citizen and 100 citizen.

4.4.3 Assessment Based Upon Data Size = 2 GB. The SpinalNet_KeyGen is evaluated by changing key length with diverse citizens is described in Figure 14. Figure 14(a) explains performance assessment of SpinalNet_KeyGen with regard

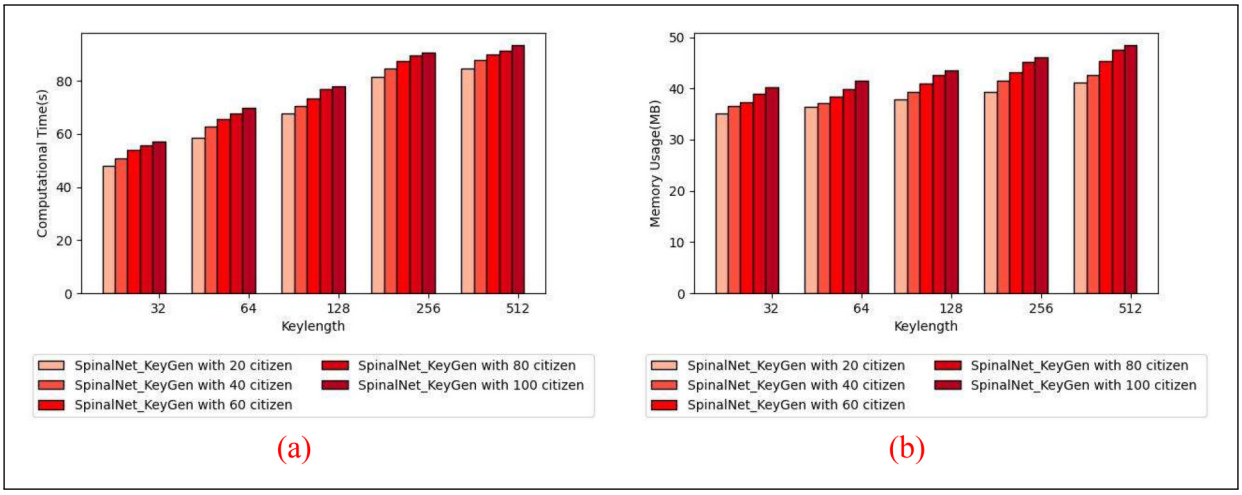

Figure 14. Assessment based upon 2 GB: (a) computational time and (b) memory usage.

to computational time. When key length is considered as 512, computational time achieved by SpinalNet_KeyGen is 84.567 s, 87.622 s, 89.756 s, 91.457 s and 93.458 s with 20 citizen, 40 citizen, 60 citizen, 80 citizen and 100 citizen. The analysis of SpinalNet_KeyGen in terms of memory usage is explicated in Figure 14(b). A Memory usage attained by SpinalNet_KeyGen is 41.1 MB with 20 citizen, 42.6 MB with 40 citizen, 45.3 MB with 60 citizen, 47.6 MB with 80 citizen and 48.5 MB with 100 citizen for key length is 512.

4.5 Comparative Techniques

SpinalNet_KeyGen is compared with few techniques reviewed like Chameleon Hashing (Ranjith Kumar & Bhalaji, 2021), EHR sharing (Wang et al., 2019), PrivChain (Malik et al., 2022), U-DBFT (Boualouache et al., 2021) and Consortium blockchain-based privacy-preserved data (CD-BPPD) to show an effectiveness of SpinalNet_KeyGen.

4.6 Comparative Analysis

SpinalNet_KeyGen is assessed based on 1 GB, 1.5 GB and 2 GB by varying keylength with consideration of performance measures.

4.6.1 Assessment Based Upon Data Size = 1 GB. An analysis of SpinalNet_KeyGen with respect to metrics by varying key length is interpreted in Figure 15. An estimation of introduced SpinalNet_KeyGen based on computational time is demonstrated in Figure 15(a). SpinalNet_KeyGen acquired computational time of 37.526 s for key length = 512 while computational time obtained by existing techniques are 90.912 s, 78.090 s, 65.447 s, 52.924 s and 40.422 s. Figure 15(b) signifies an assessment of SpinalNet_KeyGen with regard to memory usage. When key length = 512, memory usage attained by designed SpinalNet_KeyGen is 37.9 MB whereas memory usage acquired by Chameleon Hashing is 40.8 MB, EHR sharing is 40.6 MB, PrivChain is 39.8 MB, U-DBFT is 39.4 MB and CD-BPPD is 38.8 MB.

4.6.2 Assessment Based Upon Data Size = 1.5 GB. Figure 16 interprets comparative analysis of SpinalNet_KeyGen regarding measures for evaluation by altering key length. The assessment of designed SpinalNet_KeyGen considering computational time is shown in Figure 16(a). Computational time achieved by SpinalNet_KeyGen is 56.483 s where comparative techniques obtained 177.422 s, 147.931 s, 120.814 s, 89.230 s and 61.051 s for key length = 512. Figure 16(b) demonstrates an evaluation of SpinalNet_KeyGen with respective to memory usage. When key length = 512, memory usage achieved by SpinalNet_KeyGen is 43.3 MB while memory usage acquired by Chameleon Hashing is 47.2 MB, EHR sharing is 46.7 MB, PrivChain is 46.2 MB, U-DBFT is 45.7 MB and CD-BPPD is 45.2 MB.

4.6.3 Assessment Based Upon Data Size = 2 GB. An assessment of SpinalNet_KeyGen considering metrics for evaluation by varying key length is shown in Figure 17. The analysis of SpinalNet_KeyGen with regard to computational time is illustrated in Figure 17(a). SpinalNet_KeyGen obtained computational time of 94.546 s for key length = 512 whereas computational time obtained by comparative methods are 192.453 s, 166.452 s, 155.427 s, 129.753 s and 101.453 s.

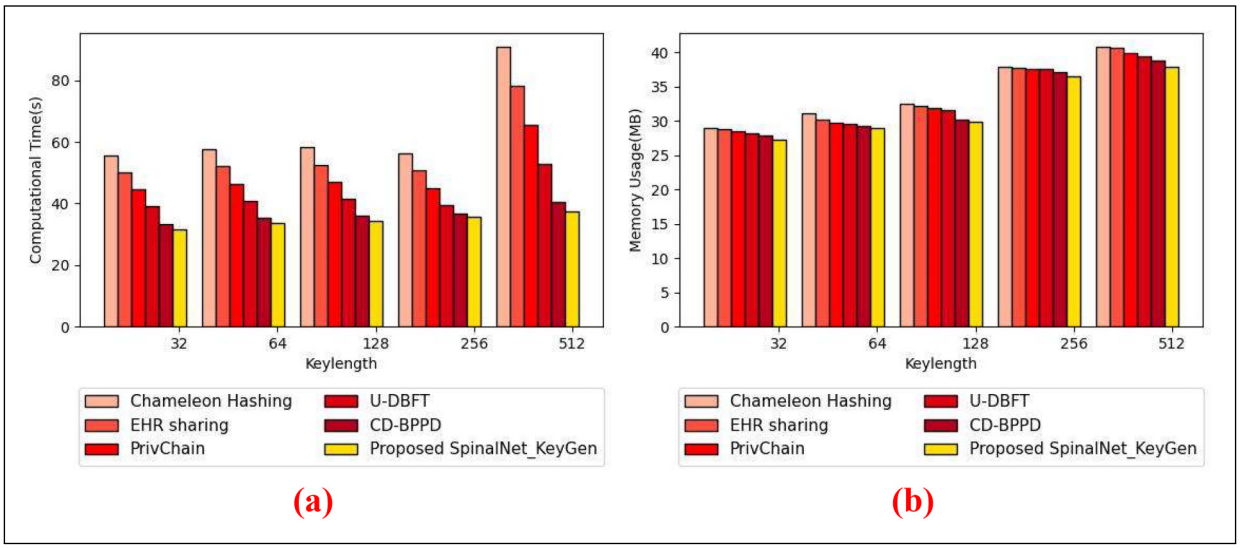

Figure 15. Comparative assessment based upon 1 GB: (a) computational time and (b) memory usage.

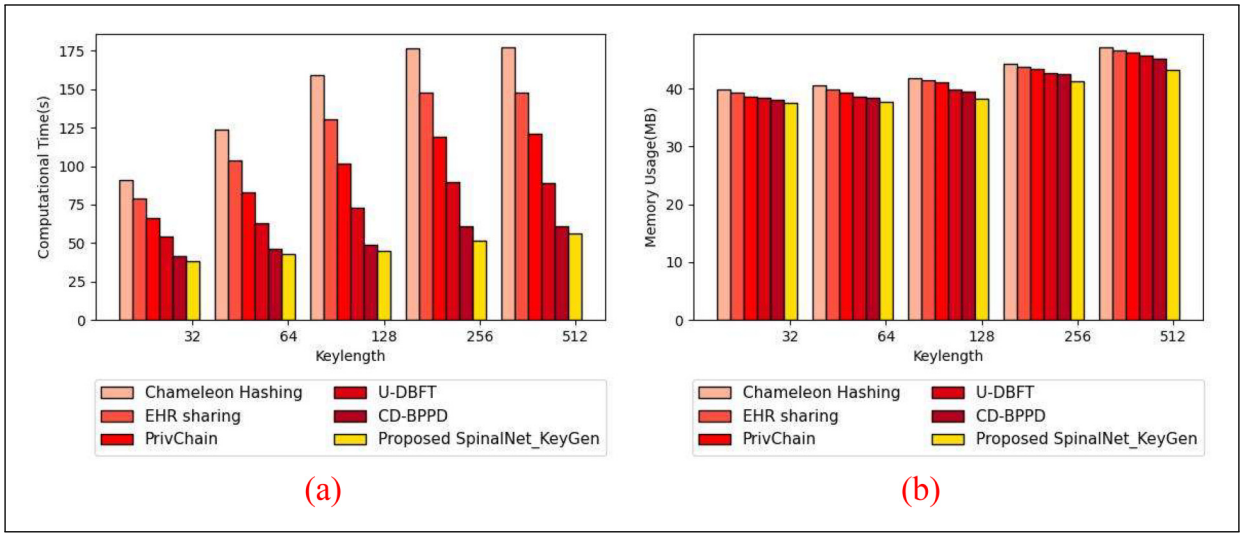

Figure 16. Comparative assessment based upon 1.5 GB: (a) computational time and (b) memory usage.

Figure 17(b) specifies an estimation of SpinalNet_KeyGen based on memory usage. When the key length is considered as 512, memory usage acquired by newly designed SpinalNet_KeyGen is 47.9 MB whereas memory usage attained by Chameleon Hashing is 56.4 MB, EHR sharing is 54.6 MB, PrivChain is 53.2 MB, U-DBFT is 52.7 MB and CD-BPPD is 49.7 MB.

4.7 Comparative Discussion

SpinalNet_KeyGen designed for PPDS in an e-governance system attained better results while comparing with traditional approaches and the results achieved are discussed in Table 3. This shows clearly that SpinalNet_KeyGen obtained minimum computational time and memory usage of 37.526 s and 37.9 MB when considering data size = 1 GB for key length = 512. The SpinalNet acquires input in gradual manner and repeatedly. Also, the individual layer of SpinalNet contributes against local outputs. Moreover, it transfers the transformed version of the inputs against global outputs. Thus, the performance of the devised model is improved than other existing methods.

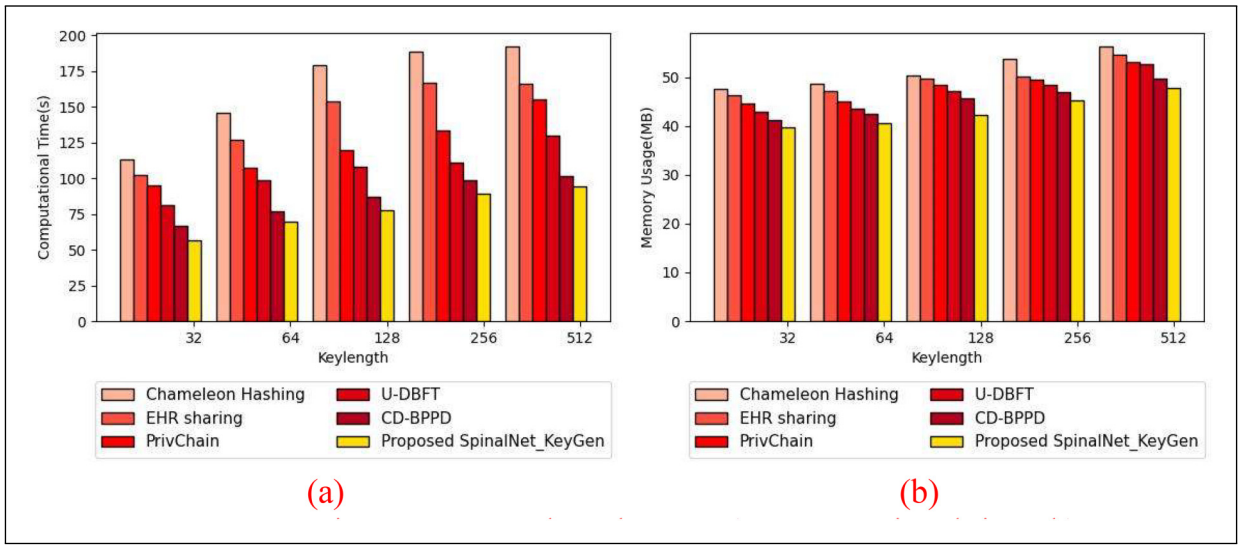

Figure 17. Comparative assessment based upon 2 GB: (a) computational time and (b) memory usage.

Table 3. Comparative Discussion of SpinalNet_KeyGen.

Analysis based upon	Metrics/ methods	Chameleon hashing	EHR sharing	PrivChain	U-DBFT	CD-BPPD	Proposed SpinalNet_KeyGen
Data size = 1 GB	Computational time (s)	90.912	78.090	65.447	52.924	40.422	37.526
	Memory usage (MB)	40.8	40.6	39.8	39.4	38.8	37.9
Data size = 1.5 GB	Computational time (s)	177.422	147.931	120.814	89.230	61.051	56.483
	Memory usage (MB)	47.2	46.7	46.2	45.7	45.2	43.3
Data size = 2 GB	Computational time (s)	192.453	166.452	155.427	129.753	101.453	94.546
	Memory usage (MB)	56.4	54.6	53.2	52.7	49.7	47.9

5 Conclusion

The present e-government methods are centralized and hence subjected to single point failures. In this research, the entities such as private agency, support GA, citizen, and GA are considered. The model comprises of service access, consortium blockchain, network layer and ledger storage. The steps carried out in this work are initialization, delegates and witness voting, new node creation, key generation, user registration, authentication, validation, data protection and finally data sharing. A data protection stage is performed using data transformation, XOR operation, encryption, hashing function and interpolation. In key generation, secret key is generated by employing SpinalNet. The designed scheme follows communication in three groups like G2C, G2G and G2B. Additionally, SpinalNet_KeyGen acquired minimum computational time and minimum memory usage of 37.526 s and 37.9 MB. The proposed method is used to improve the government's capability for simplifying every process. However, some security parameters are not considered in this work. In the future task, practical privacy and security techniques will be devised in development of blockchain and its applications. Also, the case studies will be conducted for demonstrating the successful implementation of the devised approach.

Acknowledgements

The authors would like to express their very great appreciation to the co-authors of this manuscript for their valuable and constructive suggestions during the planning and development of this research work.

Statements and Declarations

Ethical Approval

Not applicable.

Informed consent

Not applicable.

Author Contributions

All authors have made substantial contributions to conception and design, revising the manuscript, and the final approval of the version to be published. Also, all authors agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Data availability statement

None.

References

Assiri, H., Nanda, P., & Mohanty, M. (2020). Secure e-governance using blockchain. *EasyChair Preprint*, *4252*, 1–7. https://cognizium.io/uploads/resources/NA%20-%20Securing%20e-Governance%20Using%20Blockchain%20-%202017%20-%20Sep%20-%20Paper.pdf

Bannister, F., & Connolly, R. (2012). Defining e-governance. *E-Service Journal: A Journal of Electronic Services in the Public and Private Sectors*, *8*(2), 3–25. https://doi.org/10.2979/eservicej.8.2.3

Boualouache, A., Sedjelmaci, H., & Engel, T. (2021). Consortium blockchain for cooperative location privacy preservation in 5G-enabled vehicular fog computing. *IEEE Transactions on Vehicular Technology*, *70*(7), 7087–7102. https://doi.org/10.1109/TVT.2021.3083477

Chen, T., Yu, Y., Duan, Z., Gao, J., & Lan, K. (2020). Blockchain/ABE-based fusion solution for E-government data sharing and privacy protection. In Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering, pp. 258–264.

Dai, Y., Xu, D., Zhang, K., Maharjan, S., & Zhang, Y. (2020). Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks. *IEEE Transactions on Vehicular Technology*, *69*(4), 4312–4324. https://doi.org/10.1109/TVT.2020.2973705

Dipu Kabir, H. M., Abdar, M., Jalali, S. M. J., Khosravi, A., Atiya, A. F., Nahavandi, S., & Srinivasan, D. (2022). Spinalnet: Deep neural network with gradual input. *IEEE Transactions on Artificial Intelligence*, *4*(5), 1165–1177. https://doi.org/10.1109/TAI.2022.3185179

Dorri, A., Steger, M., Kanhere, S. S., & Jurdak, R. (2017). Blockchain: A distributed solution to automotive security and privacy. *IEEE Communications Magazine*, *55*(12), 119–125. https://doi.org/10.1109/MCOM.2017.1700879

Dwork, C. (2006). Differential privacy. In Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10–14, 2006, Proceedings, Part II 33, (pp. 1–12). Berlin, Heidelberg: Springer

Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4–7, 2006. Proceedings 3, (pp. 265–284). Berlin, Heidelberg: Springer.

Elisa, N., Yang, L., Chao, F., & Cao, Y. (2018). A framework of blockchain-based secure and privacy-preserving E-government system. *Wireless Networks*, *29*, 1–11. https://doi.org/10.1007/s11276-018-1883-0

Gomez Marmol, F., Sorge, C., Ugus, O., & Perez, G. M. (2012). Do not snoop my habits: Preserving privacy in the smart grid. *IEEE Communications Magazine*, *50*(5), 166–172. https://doi.org/10.1109/MCOM.2012.6194398

Guo, Y., Xi, Y., Wang, H., Wang, M., Wang, C., & Jia, X. (2024). FedEDB: Building a federated and encrypted data store via consortium blockchains. *IEEE Transactions on Knowledge and Data Engineering*, *36*(11), 6210–6224. https://doi.org/10.1109/TKDE.2023.3341149

Jain, M., & Jalia, M. (2021). Block chain in privacypreservation of records in education—A current research trend. *Psychology and Education*, *58*(1), 5300–5306. https://doi.org/10.17762/pae.v58i1.2119

Khowaja, S. A., Khuwaja, P., Dev, K., Lee, I. H., Khan, W. U., Wang, W., Qureshi, N. M. F., & Magarini, M. (2023). A secure data sharing scheme in community segmented vehicular social networks for 6G. *IEEE Transactions on Industrial Informatics*, *19*(1), 890–899. https://doi.org/10.1109/TII.2022.3188963

Li, H., Shi, D., Wang, W., Liao, D., Gadekallu, T. R., & Yu, K. (2022). Secure routing for LEO satellite network survivability. *Computer Networks*, *211*, 109011. https://doi.org/10.1016/j.comnet.2022.109011.

Lim, S. Y., Fotsing, P. T., Almasri, A., Musa, O., Kiah, L. M., Ang, T. F., & Ismail, R. (2018). Blockchain technology the identity management and authentication service disruptor: A survey. *International Journal on Advanced Science, Engineering and Information Technology*, *8*(4-2), 1735–1745. https://doi.org/10.18517/ijaseit.8.4-2.6838

Lu, R., Liang, X., Li, X., Lin, X., & Shen, X. (2012). EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications. *IEEE Transactions on Parallel and Distributed Systems*, *23*(9), 1621–1631. https://doi.org/10.1109/TPDS.2012.86

Malik, S., Dedeoglu, V., Kanhere, S. S., & Jurdak, R. (2022). Privchain: Provenance and privacy preservation in blockchain enabled supply chains. In Proceedings of 2022 IEEE international conference on blockchain (Blockchain), pp. 157–166.

Mandala, J., & Chandra Sekhara Rao, M. V. P. (2019). HDAPSO: Enhanced privacy preservation for health care data. *Journal of Networking and Communication Systems*, *2*(2), 10–19. https://doi.org/10.46253/jnacs.v2i2.a2

Mukhopadhyay, U., Skjellum, A., Hambolu, O., Oakley, J., Yu, L., & Brooks, R. (2016). A brief survey of cryptocurrency systems. In Proceedings of 2016 14th annual conference on privacy, security and trust (PST), pp. 745–752.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, 21260. https://bitcoin.org/bitcoin.pdf

Omori, Y., & Yamashita, T. (2020). Extended inter-device digital rights sharing and transfer based on device-owner equality verification using homomorphic encryption. *IEICE Transactions on Information and Systems*, *103*(6), 1339–1354. https://doi.org/10.1587/transinf.2019EDP7163

Piao, C., Hao, Y., Yan, J., & Jiang, X. (2021). Privacy protection in government data sharing: An improved LDP-based approach. *Service Oriented Computing and Applications*, *15*(4), 309–322. https://doi.org/10.1007/s11761-021-00315-3

Ranjith Kumar, M. V., & Bhalaji, N. (2021). Blockchain based chameleon hashing technique for privacy preservation in E-governance system. *Wireless Personal Communications*, *117*(16), 987–1006. https://doi.org/10.1007/s11277-020-07907-w.

Sei, Y., Okumura, H., Takenouchi, T., & Ohsuga, A. (2017). Anonymization of sensitive quasi-identifiers for l-diversity and t-closeness. *IEEE Transactions on Dependable and Secure Computing*, *16*(4), 580–593. https://doi.org/10.1109/TDSC.2017.2698472

Shailaja, G. K., & Guru Rao, C. V. (2019). Impact of opposition intensity on improved cuckoo search algorithm for privacy preservation of data. *Journal of Networking and Communication Systems*, *2*(4), 33–41. https://doi.org/10.46253/jnacs.v2i4.a4

Shen, J., Liu, D., Sun, X., Wei, F., & Xiang, Y. (2020). Efficient cloud-aided verifiable secret sharing scheme with batch verification for smart cities. *Future Generation Computer Systems*, *109*, 450–456. https://doi.org/10.1016/j.future.2018.10.049

Shyamala Susan, V., & Christopher, T. (2016). Anatomisation with slicing: A new privacy preservation approach for multiple sensitive attributes. *SpringerPlus*, *5*(1), 1–21. https://doi.org/10.1186/s40064-015-1659-2.

Srinivas, D. B., & Mohan, S. (2022). Anonymized network monitoring for intrusion detection systems. *International Journal of Computer Science and Network Security*, *22*(7), 191–198. https://doi.org/10.22937/IJCSNS.2022.22.7.23

Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557–570. https://doi.org/10.1142/S0218488502001648

Uma Maheswari, J., Somasundaram, S. K., & Sivakumar, P. (2024). Hybrid optimization enabled secure privacy preserved data sharing based on blockchain. *Wireless Networks*, *30*, 1553–1574. https://doi.org/10.1007/s11276-023-03588-y

Vu, D.-H., Luong, T.-D., & Ho, T.-B. (2020). An efficient approach for secure multi-party computation without authenticated channel. *Information Sciences*, *527*, 356–368. https://doi.org/10.1016/j.ins.2019.07.031

Wang, W., Huang, H., Yin, Z., Gadekallu, T. R., Alazab, M., & Su, C. (2023). Smart contract token-based privacy-preserving access control system for industrial internet of things. *Digital Communications and Networks*, *9*(2), 337–346. https://doi.org/10.1016/j.dcan.2022.10.005

Wang, W., Huang, H., Zhang, L., Han, Z., Qiu, C., & Su, C. (2021). BlockSLAP: Blockchain-based secure and lightweight authentication protocol for smart grid. The proceeding of IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Guangzhou, China.

Wang, Y., Zhang, A., Zhang, P., & Wang, H. (2019). Cloud-assisted EHR sharing with security and privacy preservation via consortium blockchain. *IEEE Access*, *7*, 136704–136719. https://doi.org/10.1109/ACCESS.2019.2943153

Yang, Y., Yang, Y., Guo, D., Wang, W., Nie, J., Xiong, Z., Xu, R., & Zhou, X. (2023). Stochastic geometry-based age of information performance analysis for privacy preservation-oriented Mobile crowdsensing. *IEEE Transactions on Vehicular Technology*, 1–14. https://doi.org/10.1109/TVT.2023.3252167

Zhang, L., Peng, M., Wang, W., Jin, Z., Su, Y., & Chen, H. (2021). Secure and efficient data storage and sharing scheme for blockchain-based mobile-edge computing. *Transactions on Emerging Telecommunications Technologies*, *32*(5), 110–116. https://doi.org/10.1002/ett.4315