Xiao *et al. J Wireless Com Network* (2021) 2021:165
https://doi.org/10.1186/s13638-021-02037-8

EURASIP Journal on Wireless Communications and Networking

RESEARCH

Open Access

# A blockchain-based privacy-preserving 5G network slicing service level agreement audit scheme

Ke Xiao, Ziye Geng, Yunhua He*, Gang Xu, Chao Wang and Yutong Tian

*Correspondence:
heyunhua@ncut.edu.cn
School of Information Science and Technology, North China University of Technology, Beijing, China

**Abstract**

Network slicing, as a key technique of 5G, provides a way that network operators can segment multiple virtual logic on the same physical network and each customer can order specific slicing which can meet his requirement of 5G service. The service level agreement of network slicing (NS-SLA) of 5G, as a business agreement signed between the network operators and the customers, specifies the relevant requirements for the 5G services provided by the network operators. However, the authenticity of auditing results may not be guaranteed and the customer's data may be leaked in the existing NS-SLA audit scheme. In this paper, a blockchain-based 5G network slicing NS-SLA audit model is proposed to address the above problems. The blockchain is applied as a public platform and all the dual monitored service parameters will be stored on the blockchain to ensure the authenticity of data. A trapdoor order-revealing encryption algorithm is introduced to audit strategy, which can encrypt the monitored parameters, realize the comparison over ciphertexts and prevent the privacy of data from leaking. Besides, an NS-SLA audit smart contract is designed to implement the audit task and execute corresponding punishment strategies automatically. We make experiments to exam the cost of the blockchain-based system and the results found clear support for the feasibility of the proposed model.

**Keywords:** 5G, Blockchain, SLA, QoS

## 1 Introduction

Network slicing, as a key technique of 5G, can divide a single physical network into multiple isolated logical networks to support services with a diverse set of performance and service requirements [1]. Different types of network slices can provide services with different requirements [2]. For example, services with high capacity, services with exponentially low latency, services with large-scale connections, and so on [2]. After publishing 5G industry standards, 5G network operators, such as CMCC, CUCC, and T-Mobile, have deployed 5G equipment and provided 5G network slicing services with different requirements for network performance and functions to seize the market.

Although 5G services are developing well, the network slicing may not be possible to satisfy the quality of service (QoS) requirements of different customers all the time and some conflicts may occur between network operators and customers [3]. In 2019, South Korea's three major network operators companies KT, SK, and LG Uplus all launched 5G mobile phone network services [4]. Customers using the 5G service reported that it is difficult for their devices to search for 5G signals, and the 5G network is extremely unstable. In the USA, AT&T and T-Mobile provide low-band 5G services in many different regions and states [5]. However, customers complained that the speed improvement seen in low-band 5G is minimal and it may not be worth it for customers to upgrade to 5G phones if only low-band is available. According to data from Opensignal which tracks the speed of global wireless networks [6], 5G download speeds in the USA are only 1.8 times faster than 4G LTE in 2020 which cannot meet the requirement of customers.

The service level agreement of network slicing (NS-SLA), as an agreement signed by the network operators and the customers [7], is proposed to address the above issues. The NS-SLA determines the requirement of network services performance and maps the requirement of QoS to the service attribute parameter level. The network operators customize network slicing according to the specific requirements in the NS-SLA reached between the customer and the operator to provide corresponding services. Each specific NS-SLA between a customer and a network operator has formulated different parameter requirements, such as secrecy rate, latency, packet loss rate, and so on. Thus, how to ensure that SLA is executed is crucial to ensuring service quality.

Many scholars have done a lot of research about SLA monitoring and auditing in different domains, such as Cloud platforms and Web service. Traditional SLA monitoring schemes [8] depend on third-party auditors (TPA) who may tamper with the report for benefits. Although multi-party monitoring schemes [9, 10] can address the above issue, they are difficult to establish a trust relationship between multiple parties, and the authenticity of data cannot be guaranteed. Blockchain [11], as a trustful platform, can establish a trust relationship between multiple parties [12]. The blockchain technique is introduced into the SLA monitoring and auditing [13, 14], however, the privacy leakage issue in audit task still remains a challenge. Compared with Cloud platforms and Web service, there is relatively little research on SLA management of 5G network slicing service. Some researchers [15] design a SLA Manager to connect with the monitoring module and audit service parameters, however, the authenticity of parameters cannot be guaranteed and these schemes don't consider the privacy leakage issue. Lots of privacy protection methods are proposed, for example, differential privacy technique, order-preserving encryption (OPE) [16], order-revealing encryption (ORE) [17, 18], and so on. However, a privacy protection method suit for network slicing SLA audit needs to be explored.

To address the above issues, we propose a blockchain-based 5G network slicing NS-SLA audit model in the existence of eavesdroppers. In this model, a customer can submit own NS-SLA requirements to order a specific network slice from network operators through CSMF (communication service management function). Both the customers and the network operators are involved in monitoring the task of 5G network slicing service and upload the monitored data to the blockchain. The blockchain provides a public and transparent platform to ensure the recorded data cannot be tampered. Since the monitored data may leak the privacy, the order-revealing encryption scheme (TORE) is introduced to encrypt the parameters and provide a way that realizes ciphertexts comparison.

Besides, an NS-SLA audit smart contract is designed to perform the audit task and punish the offending party automatically. The contributions of this paper are summarized as follows:

- We present a blockchain-based 5G network slicing NS-SLA audit scheme which realizes credible auditing and ensures the immutability of the recorded parameters on the blockchain.
- The TORE algorithm-based privacy-preserving scheme is proposed to ensure the security of the monitored parameters and realizes the ciphertext auditing. All the audit tasks can be performed automatically by the designed NS-SLA audit smart contract.
- We evaluate the cost of the proposed audit model which demonstrates the feasibility of the model.

The rest of this paper is organized as follows. In Sect. 2, we present the blockchain-based 5G NS-SLA audit model. The detailed TORE-based privacy-preserving NS-SLA audit is presented in Sect. 3. In Sect. 4, we conduct the performance evaluation of the proposed model. At last, the paper is concluded in Sect. 5.

## 2 Related work

### 2.1 SLA monitoring and auditing

Many scholars have done a lot of research about SLA management in different domains, such as Cloud platforms and Web service. For example, traditional SLA monitoring schemes depend on service providers [19] or third-party auditors (TPA) [8] to monitor service. However, all these parts may tamper with the report for benefits. Some multi-party monitoring schemes based on reputation mechanisms [9] or reasoning techniques [10] are proposed to improve SLA monitoring by collecting and inferring information from different SLA parameters. However, it is difficult to establish a trust relationship between multiple parties, and either party may provide false data. Blockchain brings new opportunities [11, 20], it is a distributed ledger that implements consensus mechanism to ensure data consistency [12] and establishes a trust relationship between multiple parties. Thus, blockchain may provide a trustful platform for SLA monitoring and auditing [14, 21]. In the blockchain-based SLA auditing scheme [13], all witnesses which are selected from the blockchain network nodes can detect violation and report credible feedback cooperatively. However, the ability of witness is limited and user's privacy may be disclosed due to the openness and transparency of blockchain.

Similar to the Cloud platforms [22] and Web service [23], the 5G network slicing service also needs SLA to manage the service. Some researchers [24] proposed mapping mechanisms for slice template generation according to customers' information provided in the SLA. However, whether the service quality meets the SLA requirements is not considered. Papageorgiou et al. [15] designed a SLA Manager. The SLA manager is connected with the monitoring module to receive the real-time value of the current monitoring indicators and verify whether the service parameters are within the range of user requirements. However, the SLA Manager cannot guarantee the authenticity of the source of monitoring data. Touloupou et al. [25] designed a monitoring system that can automatically develop agreements based on each network slice instance and detect any violations and raise alerts. However, the proposed framework can only monitor and manage business assurance in a single network service. All these schemes don't consider the privacy issue during SLA management. Liu et al. [26] proposed a secure federated learning framework that uses local differential privacy technique and adds Gaussian noise to avoid customer privacy leakage in 5G networks. Kang et al. [27] proposed a Proof-of-Verifying consensus scheme to defend against poisoning attacks and ensure data security. Although there are lots of privacy protection method, how to design a privacy protection method suit for network slicing SLA audit needs to be explored.

### 2.2 Order-revealing encryption

Since order-preserving encryption (OPE) [16] can ensure that the sort order of ciphertexts is consistent with the corresponding plaintext sorting order, it is considered as one of the feasible solutions to solve the privacy leakage problem in SLA monitoring and audit. However, Naveed and Kamara [28] proved that databases encrypted by OPE are exceedingly vulnerable to "inference attacks." The order-revealing encryption (ORE) as an improved OPE has been proposed [29]. The plaintext comparison result can be determined by ciphertext and corresponding indexes in the ORE scheme [17]. However, the leaked information in the scheme may be caused by the different indexes of the first bit of the two encrypted data. Lewi and Wu [18] proposed a trapdoor order-revealing encryption scheme (TORE) which disclosures less sequential information and provides stronger data security.

## 3 Smart contract-based 5G SLA auditing model

There are three major scenarios of 5G network, including ultra-reliability and low-latency communication (uRLLC), massive machine-type communication (mMTC), enhanced mobile broadband (eMBB). The customer can order specific slicing which can meet his requirement under different scenarios. Since the uRLLC scenarios require ultra-real-time and high reliable services, the mMTC scenarios require massive connections, and the eMBB scenarios demand ultra-high bandwidth and big data services, a network slicing under different scenarios focus on different requirements of service parameters, as shown in Table 1.

In this section, we consider the scenario consisting of customers, network operators, base stations (BS), and some eavesdroppers. Besides, a blockchain-based 5G NS-SLA audit system model is elaborated. In this scenario, the network operators can provide customers with customized network slicing with different QoS requirements. The slicing network architecture with BS deployment for end-to-end network slicing is shown in Fig. 1. The network operator is responsible for maintaining the network.

The customer will order a network slicing with the satisfied requirement in NS-SLA from a network operator. This NS-SLA maps the requirement of QoS to the service attribute parameter level, each specific NS-SLA can be extended or customized standard service parameters, such as secrecy rate, delay, packet loss rate, and so on. In order to audit whether the QoS meets the NS-SLA standard, both customers and network operators will monitor the specific 5G service parameters. These service parameters will be recorded on the blockchain to keep the authenticity of data. Due to the transparency of

**Table 1** Network slicing SLA parameters

| Slicing type | Parameter notation | Description |
|---|---|---|
| eMBB | Peak rate | The maximum data rate that can be achieved under ideal conditions |
| | Capacity | Product of bandwidth, spectral efficiency, and number of units |
| | Mobility | The maximum relative moving speed of both communication parties under the premise of satisfying certain system performance |
| | Spectrum efficiency | Spectrum efficiency, also known as system capacity, is a measure of the effectiveness of a system and describes how much capacity can be provided |
| mMTC | Number of connections | Number of connected terminal devices per square kilometer |
| | Power consumption | Power consumption mainly refers to the power loss of communication-related equipment |
| uRLLC | Reliability | As an important index to measure the performance of communication system, reliability refers to the reliability of information received in a given channel |
| | Mobility | The maximum relative moving speed of both communication parties under the premise of satisfying certain system performance |
| | Latency | End-to-end latency is a measure of the time required to successfully receive a message from the source to the target on the communication interface |

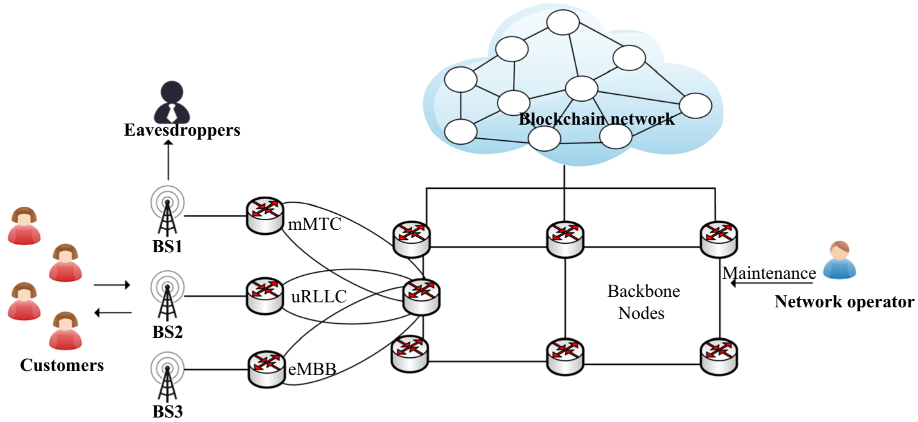

**Fig. 1** Architecture of slicing network. The model consists regional customers, network operator, and eavesdroppers. The figure shows the slicing network architecture with BS deployment for end-to-end network

blockchain, all the parameters will be encrypted before uploading it to the blockchain. In addition, a NS-SLA audit smart contract is implemented to execute audit tasks. Once violations happen, the SLA audit smart contract will be triggered to compare the monitoring data and audit data. Then, the smart contract automatically distributes rewards or punishes the offending party, with the reputations updated. The whole audit model is depicted in Fig. 2.

### 3.1 Blockchain-based system

This NS-SLA audit system is based on blockchain which is regarded as a trustful platform. In the blockchain network, servers from different network operators (as full nodes) can be registered as miners to package and execute transactions. However, if

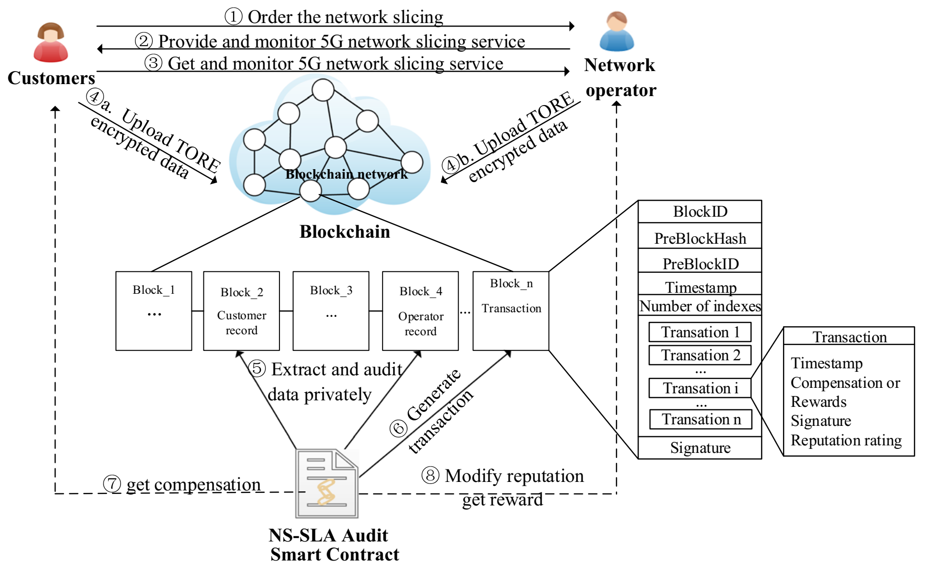

**Fig. 2** System architecture for 5G NS-SLA audit model. The model of our scheme consists regional customers and network operator. This figure shows the audit process of our scheme

more than half of the miners come from the same network operator, the provider may launch 51%-Attack. In order to prevent someone from controlling the computing power of the blockchain network, the number of miners provided by each network operator will be limited to no more than one-third of the total number of miners. In addition, customers with personal devices can register as light nodes to download block headers or view transactions.

In our blockchain-based system, customers and network operators will store their monitoring data on the blockchain. Any transaction initiated by the customer in the blockchain network will be broadcasted and verified by miners in the blockchain network. The consensus protocol of blockchain can guarantee the security and consistency of transactions. Once the transaction passes verifications, it will be permanently recorded into the blockchain and spread throughout the network which cannot be tampered. In order to monitor and verify the transaction on the blockchain, some miners in the blockchain will be selected as witnesses to finish verification tasks. The selection of witness can be executed by a witness smart contract to ensure unbiasedness and randomness of the selection process [13].

When the monitoring data has been uploaded to the blockchain, the NS-SLA audit smart contract will be triggered to finish auditing tasks. This smart contract is designed according to NS-SLA requirements and it can interact with customers and network operators. After auditing, the NS-SLA audit smart contract will automatically generate a transaction to distribute rewards and update users' reputations.

### 3.2 Audit model

The core of the model is violations audit by comparing the monitoring data from customers and network operators. Therefore, the audit model is one of the most important parts, which is described as follows.

Step 1 *Obtain monitoring service parameters* After a customer orders a network slice from a network operator, the customer will sign the NS-SLA agreement with the network operator. When the 5G network slice service starts, the network operator will continuously monitor service parameters. The monitoring parameters of the network operator are obtained by monitoring the base station to provide more reliable monitoring data. The credibility of data can be achieved by the base station in the monitoring phase. Besides, in order to ensure the authenticity of the data, the customers also monitor service parameters.

Step 2 *Encrypt and upload monitored parameters* Since all the monitoring data will be stored on the blockchain, the data needs to be encrypted to keep its security. On the network operator side, the monitoring parameters will be encrypted by the TORE, and then the encrypted data will be uploaded to the blockchain. Similar to the network operator side, the parameters monitored by customers are also encrypted by TORE, and then the corresponding ciphertexts are uploaded to the blockchain. The security of data is guaranteed by the TORE algorithm in the transmission phase. The data recorded on the blockchain is also credible, since the blockchain can record behaviors and data of the customer and network operator, which cannot be tampered.

Step 3 *Extract and audit service parameters* After storing the monitored data on the blockchain, the SLA audit smart contract extracts ciphertexts of the two sides from the corresponding block to maintain the credibility of the whole auditing process. Then, it will compare these encrypted data. If the monitoring data of both sides are equal, it proves that the true monitoring data is recorded. Then, actual monitoring data will be compared with the standard values of service parameters, to determine whether the service conforms to the 5G SLA standard.

Step 4 *Punishment* According to the audit result, the smart contract will judge the fault party and some punishments will be implemented. For example, network operators will compensate customers if the quality of service they provide does not meet the requirements of NS-SLA, and network operators will be paid if the quality of service meets the requirements.

## 4 Methods

In this section, we present the privacy audit model of 5G NS-SLA in detail. The proposed model can audit multiple service parameters according to the requirement in NS-SLA of specific network slicing. In order to audit these service parameters, the dual monitoring method is used. After that, all the monitored parameters will be encrypted by the TORE algorithm to ensure the security of data stored on the blockchain. A NS-SLA audit smart contract is designed to implement the audit tasks in the blockchain network. This smart contract will be triggered to compare the ciphertexts to fix whether the data from two sides is consistent and compare the encrypted data with the standard parameter in SLA to fix whether the data achieves the service requirement. In addition, some punishments will be automatically executed according to different situations. The whole privacy-preserving audit scheme will be described in detail as follows.

### 4.1 TORE-based privacy audit mechanism

In the proposed audit scheme, each service parameter needs to be encrypted by the TORE, each service parameter $i$ will be set a specific plaintext space $[N_i]$ before encrypting and the plaintext space has individual domain. All the standard service parameters and actual service parameters monitored by customers and network operators are included in the domain. In addition, each plaintext space of service parameter is shared by the customer and network operator. For each service parameter $j$, its corresponding plaintext space is recorded as $\left[N_j\right]$ and $\left[N_j\right]$ has $n$ elements according to SLA requirement. All the element $x_{ji} \in \left[N_j\right]$ in the plain$\left[N_j\right]$ associate with an encryption key $k_{ji}$.

For example, the secrecy rate as one of the 5G service parameters is monitored by customer and network operator. In the model of 5G service, the customer measures the downlink rate and computes the secrecy rate, and the network operator uses the uplink rate measured by the base station to compute the secrecy rate. After that, both the customer and the network operator will encrypt their own monitoring parameters by the TORE algorithm [18] and then upload the ciphertexts to the blockchain. Despite the public and transparent nature of the blockchain, the encrypted parameters stored on the block will not be disclosed. The TORE algorithm includes a left encryption algorithm $Encrypt_L$ and a right encryption algorithm $Encrypt_R$. Moreover, the ciphertext generated by TORE consists of a left component $ct_L$ and a right component $ct_R$. The whole execution process of TORE-based privacy audit scheme includes four steps: *Setup*, $Encrypt_L$, $Encrypt_R$, and *Compare*, and the detail is shown as follows:

- $Setup\left(1^{\lambda}\right)$: Firstly, a security parameter $\lambda$ will be generated. After that, for each element in plaintext spaces, key $k_{ji} \overset{R}{\leftarrow} \{0,1\}^{\lambda}$ of a secure pseudo-random function (PRF) $F : \{0,1\}^{\lambda} \times \{0,1\}^{\lambda} \rightarrow \{0,1\}^{\lambda}$ will be generated by the SLA audit smart contract. Next, sample a uniformly random permutation $\pi : \left[N_j\right] \rightarrow \left[N_j\right]$ on each plaintext space domain. The pair $\left(k_{ji}, \pi\right)$ is the secret key $sk_{ji}$. These secret keys are shared by customers and network operators.
- $Encrypt_L\left(sk, x_j\right)$: On the network operator side, the service parameters are monitored by the base station. After receiving the monitored parameter $x_j$, the network operator will first select the corresponding secret key $sk_j$ of $x_j$. Then, the permuted position $\pi\left(x_j\right)$ of $x_j$ will be computed according to permutation function, which can ensure that $x_j$ in $\left[N_j\right]$ can't be learned from others and protect data security while comparing ciphertexts. Next, the left ciphertext for $x_j$ is computed, according to $ct_{Lj} = \left(F\left(sk, \pi\left(x_j\right)\right), \pi\left(x_j\right)\right)$. In addition, the left ciphertext $ct_{Lsj}$ of each standard service parameter in SLA will also be computed, which is similar to the method of $ct_{Lj}$. And, both the $ct_{Lj}$ and $ct_{Lsj}$ will be uploaded to the blockchain.
- $Encrypt_R\left(sk, y_j\right)$: In this step, we use a hash function $H : \{0,1\}^{\lambda} \times \{0,1\}^{\lambda} \rightarrow \mathbb{Z}_3$. Besides, a comparison function $CMP\left(m_i, m_j\right)$ is set and its outputs $-1$ with $m_i > m_j$, 0 with $m_i = m_j$, and 1 with $m_i < m_j$. The right encryption is finished by customers. First, the customer will monitor the service parameters $y_j$. Next, a random nonce $r \overset{R}{\leftarrow} \{0,1\}^{\lambda}$ will be sampled. Then, the customer will compute the index $v_{jz}$ for each $z \in \left[N_j\right]$ by using specific secret key and monitored parameter, according to $v_{jz} \leftarrow CMP\left(\pi^{-1}(z), y_j\right) + H(F(k,z), r)$ $(mod\ 3)$. Finally, a group of

index $v_{j1}$, $v_{j2}$, ..., $v_{jn}$ will be obtained and the right ciphertext is represented by the tuple $ct_{Rj} = \left(r, v_{j1}, v_{j2}, \ldots, v_{jn}\right)$. The customer will upload $ct_{Rj}$ to the blockchain.

Once all the encrypted data arrive, the SLA audit smart contract will be triggered to implement the audit task. First, the ciphertexts $ct_{Lj}$, $ct_{Rj}$ will be extracted, and then the SLA audit smart contract will execute the *Compare* algorithm automatically to obtain the comparison result between network operators and customers. The *Compare* algorithm is defined as follows:

- $Compare\left(ct_{Lj}, ct_{Rj}\right)$: After extracting $ct_{Lj}$ and $ct_{Rj}$, and the SLA audit smart contract will parse $ct_{Lj} = \left(k', h\right)$ and $ct_{Rj} = \left(r, v_{j1}, v_{j2}, \ldots, v_{jn}\right)$. Thereinto, $k' = (F(k, \pi(x)), \pi(x))$ is a PRF key and $h = \pi(x)$ is a permuted position index. Then, the index of comparison between $x_j$ and $y_j$ will be computed, according to $I_j = v_j - H\left(k', r\right) \ (mod\ 3)$. $I_j$ is 0 if $x_j = y_j$; $I_j$ is 1, if $x_j < y_j$; otherwise $I_j$ is 2.

In order to verify the correctness of this algorithm, we set $ct_{Lj} \leftarrow Encrypt_L\left(sk, x_j\right)$ and $ct_{Rj} \leftarrow Encrypt_R\left(sk, y_j\right)$. The proof is shown as follows:

$$\begin{aligned} Compare(ct_L, ct_R) &= v_{jh} - H\left(k', r\right) \\ &= CMP\left(\pi^{-1}(h), y_j\right) + H(F(k, h), r) - H\left(k', r\right) \\ &= CMP\left(\pi^{-1}\left(\pi\left(x_j\right)\right), y_j\right) + H\left(F\left(k, \pi\left(x_j\right)\right), r\right) \\ &\quad - H\left(F\left(k, \pi\left(x_j\right)\right), r\right) \\ &= CMP\left(x_j, y_j\right) \in \mathbb{Z}_3. \end{aligned} \tag{1}$$

### 4.2 Punishment strategies for violations

After finishing the comparison, there will be two kinds of comparison results. One is that all the monitored parameters from network operators and consumers are equal, $\forall j$, $I_j = 0$, which means that no party reports false data. In this situation, the SLA audit smart contract be triggered to extract $ct_{Lsj}$ from the corresponding block and then implement $Compare\left(ct_{Lsj}, ct_{Rj}\right)$ to compare the service parameters with the NS-SLA standard values. Then, we can obtain a comparison index $I_{sj}$. If $I_{sj} = 0$ or 2, which proved that the quality of service reaches the NS-SLA standard, the service fees $F_s$ will be distributed to the network operator automatically by the smart contract. Otherwise, the NS-SLA standard is not reached. Thus, the deposit $F_d$ of network operator will be distributed to the miners as a reward instead of returning back, and the customer will get the service compensation $F_c$.

The other is that the monitored parameters from the above two parts are not equal, which exists at least one index that $I_j \neq 0$. Then, the NS-SLA audit smart contract will be triggered to check the comparison information in the block during the time slice $t_s$. The times of $\exists I_j \neq 0$ and the times of total comparison $num_{all}$ will be increased by one. Moreover, the mismatching comparison result from the same customer will be only recorded once to prevent the malicious users provide fake data in many times. If $num_{dif} > \lfloor num_{all}/2 \rfloor$, the network operator is cheating, the operator's deposit will not be returned as a punishment. Otherwise, the consumer reports the false service parameters in the audit task, and then, the reputation of the customer will be reduced as a punishment, according to the following formula.

$$T_x(i,t) = T_x(i,t_s) - \left( \frac{F_{\text{deposit}} - F_{\min}}{F_{\max} - F_{\min}} \right) \times (1 - T_x(i,t_s)). \qquad (2)$$

Thereinto, $T_x(i,t)$ is the reputation value of customer $x$ who requests service $i$ in time $t$ and $T_x(i,t_s)$ is the reputation value of $x$ during time slice $t_s$. $T_x(i,t_s)$ integrates users' rating as a subjective source and service quality monitoring information as an objective source, and it can change dynamically with the service preferences of customers [30]. The punishment degree is determined by the compensation ratio of negotiated compensation $F_c$, the minimum compensation $F_{\min}$ and the maximum compensation $F_{\max}$. We use a punishment degree to measure the importance of this service task and multiply it by the customer's unreliability value as the reputation reduction value. At the same time, the customer's deposit $F_d$ will be distributed to miners and the payment $F_s$ for the service will be automatically sent to the network operator. When a consumer reputation value is lower than 0, it will not be able to apply for the 5G network slicing.

## 5 Results and discussion

We implement our model on the Ethereum platform, design the NS-SLA audit smart contract by solidity programming language and deploy it on the blockchain. Because Ethereum provides a blockchain platform that allows developers to use smart contracts to develop applications on it and it is easy for researchers to develop and test prototypes on the Ethereum blockchain. In this section, we first compare the proposed blockchain-based NS-SLA audit scheme with other audit schemes to show the advantage of the proposed scheme. After defining the scenario and the service parameters, we analyze the complexity of the proposed audit model. The complexity analysis includes communication overhead, smart contract overhead, and encryption overhead. Since communication overhead refers to the number of interactions between systems, which is the same as traditional systems, we mainly analyze the overhead of additional modules including smart contracts and encryption in our solution.

### 5.1 Comparison of the existing audit methods

From Table 2, we can see the comparison results between different schemes. Although the SLA-M scheme can monitor multiple parameters, the time overhead is relatively high. The time overhead of our audit scheme based on blockchain is within acceptable range. Both SLA-ICM [14] scheme and Sec-rSLA [13] scheme are applying blockchain and design a distributed SLA audit scheme. However, the data stored on blockchain in this scheme may leak privacy. Our scheme introduces the TROE algorithm to encrypt the service parameters to ensure the security of data and realize ciphertexts comparison. Besides, a NS-SLA audit smart contract is designed to achieve perform audit tasks automatically.

### 5.2 Overhead of SLA audit smart contract

In the proposed audit model, all the audit tasks are triggered to execute by the NS-SLA audit smart contract and the execution of the smart contract is one of the main costs.

**Table 2** Comparison of SLA audit schemes

| Scheme | Distributed model | Multiple parameters | Security | Privacy protection |
|---|---|---|---|---|
| SLA-M [8] | | √ | | |
| SLA-ICM [14] | √ | | √ | |
| Sec-rSLA [13] | √ | | √ | |
| Our scheme | √ | √ | √ | √ |

In order to measure the overhead of NS-SLA audit smart contract, we deploy it on the Ethereum blockchain. Each executing step of the smart contract is depending on the state transition of interfaces. In addition, the execution of the program defined in interfaces finished by miners in the blockchain network is the main consumption of gas (a unit that can measure the workload of miners).

Figure 3 shows the gas consumption of each interface in the NS-SLA audit smart contract. Since the complexity of different interfaces is different, so the gas consumption is different and the gas consumption increases with the complexity of interfaces. Overall, the total gas consumption of the NS-SLA smart contract is acceptable.

### 5.3 Service parameter in audit model

In the proposed scheme, the customer can order customized network slicing from the network operator according to his own QoS requirement. Thus, we define a scenario and select several parameters to evaluate the audit scheme.

In this part, we take the secrecy rate as an example of service parameters, show the acquisition process of this parameter, and evaluate the secrecy rate in the presence of eavesdroppers using simulation with the following parameters. We use the cylindrical antenna array and set the number of antennas at the transmitter as 128. The antenna

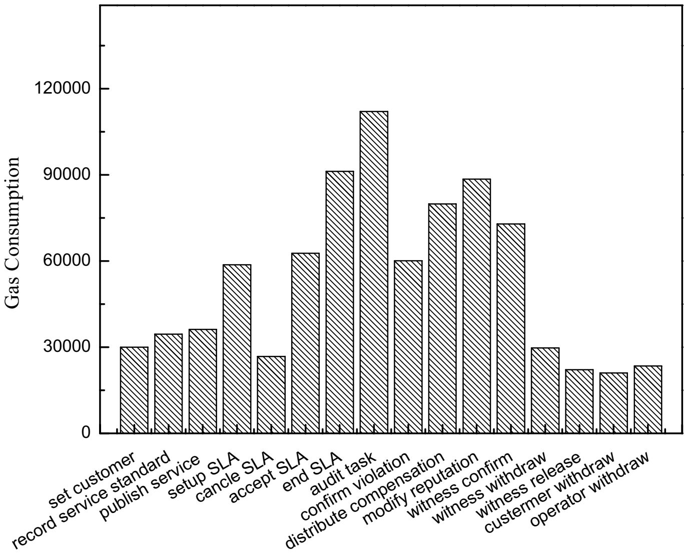

**Fig. 3** The main gas consumption of NS-SLA audit smart contract. The figure shows the gas consumption of each interface in the NS-SLA audit smart contract

layer distance is set to 0.5 wavelength, the circle radius is set to 1 wavelength, and the number of layers is set to 10. In this scenario, we consider a hybrid unicast/multicast transmission, and users are divided into multicast group (Multi-group), unicast group (BD group) and eavesdropping group (Eve group), and all group in the system (All group). The total number of users is 42, and the number of eavesdroppers is 10. The load impedance, antenna impedance, and mutual impedance are set to $Z_{\mathrm{L}} = 50\Omega$, $Z_{\mathrm{A}} = 50\Omega$, $Z_{\mathrm{M}} = 50\Omega$, respectively. Besides, the SNR is ranged from −10 to 35 dB.

In this model, the secrecy rate of a customer can be computed as follows:

$$R_{\mathrm{S}} = \max\{I_{\mathrm{D}} - I_{\mathrm{E}}, 0\}. \tag{3}$$

Thereinto, $I_{\mathrm{D}}$ is the mutual information between the legitimate senders and the legitimate receivers, which is shown as 4. $I_{\mathrm{E}}$ is the mutual information between the legitimate senders and the eavesdroppers, which is shown as 5.

$$I_{\mathrm{D}} = \log\left(1 + SINR_{m,j}\right) \tag{4}$$

$$I_{\mathrm{E}} = \log\left(1 + SINR_{N,j}\right). \tag{5}$$

Both $I_{\mathrm{D}}$ and $I_{\mathrm{E}}$ are affected by the signal interference noise ratio (SINR). The SINR of customers in the eavesdropping group is shown in formula 6, and the parameters in this formula are the same as those in our previous work [31].

$$\mathrm{SINR}_{M+1,j} = \frac{p_{M+1}\left|\hat{\mathbf{g}}_{M+1,j}^{H}\omega_{M+1}s_{M+1}\right|^2}{\sum_{n\neq M+1}^{M+1} p_n\left|\hat{\mathbf{g}}_{M+1,j}^{H}\omega_n s_n\right|^2 + \sigma^2}. \tag{6}$$

In order to maximize the secrecy rate $R_{\mathrm{S}}$, we should maximize $I_{\mathrm{D}}$ and minimize $I_{\mathrm{E}}$ as much as possible, according to formula 3.

Then, we test the secrecy rate of the BD group, multi-group, Eve group, and All group with the increase in the number of eavesdroppers. In the NS-SLA audit model, both customers and network operator will monitor the secrecy rate as a service parameter. Figure 4 illustrates that the secrecy rate of the eavesdropping group increases greatly with the increase in a number of eavesdroppers. Besides, the secrecy rate of the BD group and All group decreases greatly with the increase in the secrecy rate of the Eve group.

Then, we set secrecy rate in NS-SLA as 90 bps/Hz and the parameters monitored by customers and network operators are randomly set ranging from 80 to 120 bps/Hz according to Fig. 4 for testing the proposed audit scheme. When the monitored parameters of the customer and the network operator are the same, the NS-SLA smart contract will compare the monitored parameters with the standard parameters in NS-SLA. When the monitored parameters are not greater than 90 bps/Hz, the network operator will receive the rewards and the reputation value of customers and network operators will increase. Otherwise, the QoS of network slicing is not up to standard, the customer will get compensation. When the monitored parameters of the customer and the network operator are different, the offending party is found

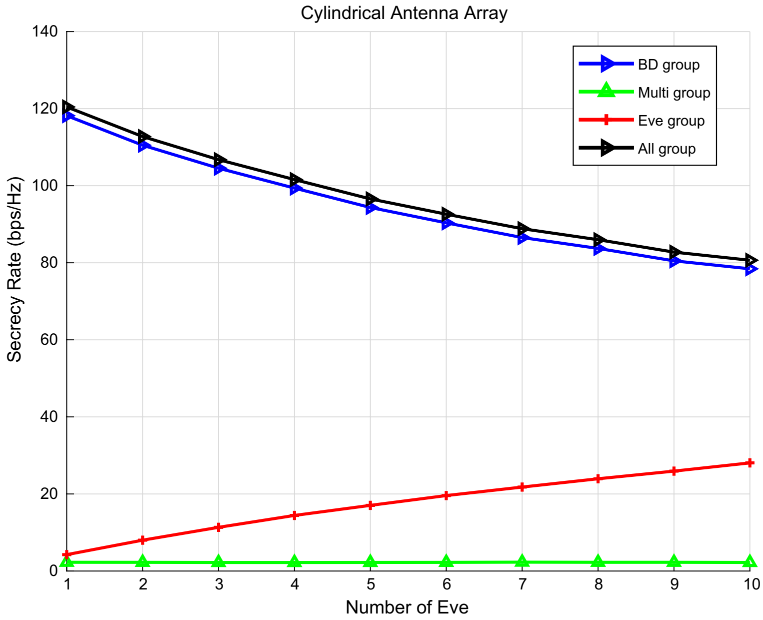

**Fig. 4** Secrecy rate under cylindrical antenna array. This figure shows the secrecy rate yielded when the antenna array is cylindrical antenna array, where the number of Eves ranges from 1 to 10. We evaluate the performances of four groups including BD group, Multi-group, Eve group, and All group

**Table 3** The time cost of encrypt and compare phase

| Plaintext space | Number of *n* | Encrypt time | Compare time |
|---|---|---|---|
| 32bit | 1 | 58.92 $\mu s$ | 0.71 $\mu s$ |
| | 2 | 110.42 $\mu s$ | 1.31 $\mu s$ |
| | 3 | 165.12 $\mu s$ | 1.90 $\mu s$ |
| | 4 | 224.01 $\mu s$ | 2.62 $\mu s$ |
| | 5 | 297.04 $\mu s$ | 3.62 $\mu s$ |

according to the monitored records. The offending party will be punished and his reputation will decline.

### 5.4 Overhead of TORE encryption

In this proposed audit model, the service parameter can be customized according to the requirement of customers. Thus, the conditions such as the number of parameters and the size of the space occupied by the parameters can be changed. Since encryption is one of the key points of the privacy audit model, we focus on the complexity analysis of the encryption algorithm. We set the number of input parameters to $n$, $l$ is the number of bits for a parameter, and $N$ is the size of the plaintext space.

The *Encrypt* phase and *Compare* phase are the main phases of the TORE algorithm, we evaluate the cost of the two phases, respectively, with the following parameters. The security parameter is $\lambda = 128$ and $F$ is instantiated with AES-128. The number $n$ of parameters is set from 1 to 5 in turn, the parameters are randomly selected, such as secrecy rate, online users *ID*, latency, and so on. Besides, the plaintext space $N$ for each parameter is set to 32 bits. Table 3 shows that the time cost of encryption is much higher than that of comparison under different numbers of parameters. Compared with the time cost of encryption, the time cost of comparison time can be negligible. Besides, the encryption time increases with the increasing number of the monitored parameters. Since the encryption time is microsecond level, the time cost is within an acceptable range.

Since the main time cost is the encryption process, we select 3 parameters including the ID of online user, secrecy rate, and latency as auditing parameters to test the encryption time with different-sized plaintext space. Different types of base stations can access different numbers of users at the same time, ranging from hundreds to tens of thousands. The requirement of secrecy rate and latency are also different in different scenarios. From Fig. 5, we can see the average encryption time of the three parameters with different sizes of plaintext space from 8 bits to 64 bits. For example, the precision secrecy rate in 32 bits plaintext space and 64 bits plaintext space are different. Thus, the precision of service parameters should be changed by expanding the plaintext space. With the size of the plaintext space becomes larger, the range of requirement of service parameter becomes wider. During each encryption process, $Encrypt_L$ will loop $n$ times making one pseudo-random permutation call and one pseudo-random function calls each iteration. $Encrypt_R$ will compare the given parameter value to $2^l$ possible values and it will loop $n2^l$ times making one hash function call, one pseudo-random permutation call, and one pseudo-random function call each iteration. The more comparison elements in plaintext space, the more index of each element will be generated in process of right encryption. $Compare\left(ct_{Lj}, ct_{Rj}\right)$ makes $n$ calls to hash. With the increase in $n$ and $l$, the encryption time will increase. Besides, the time complexity of encryption is exponential in the $l$ size.

In addition, the plaintext space of size $N$ for each service parameter will affect the length of the ciphertexts. In the left ciphertexts, $ct_L$ consists of a pseudo-random function key and a permuted position index, occupying $\lambda + \lceil \log N \rceil$ bits of space. In the right ciphertexts, $ct_R$ consists of a nonce, together with $N$ elements in $\mathbb{Z}_3$, occupying

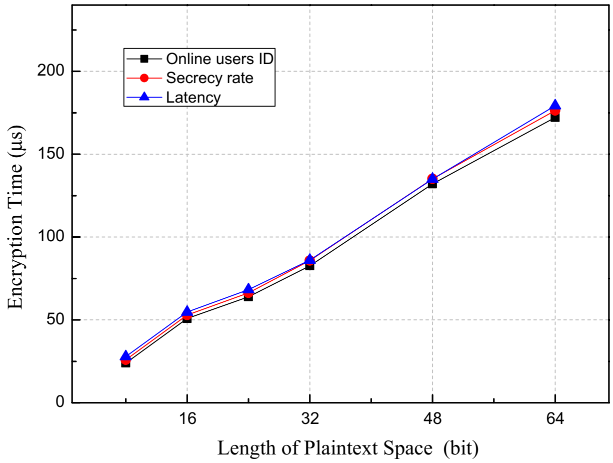

**Fig. 5** Encryption time cost of multiple parameters in different sizes of plaintext space. This figure shows the encryption time yielded, where the length of plaintext spaces are 8, 16, 32, 48, 64. We evaluate the performances of three parameters including online user ID, secrecy rate, and latency

$\lambda + \lceil N \log_2 3 \rceil$ bits of space. Thus, a whole ciphertext occupies $2\lambda + \lceil \log N \rceil + \lceil N \log_2 3 \rceil$ bits of space. For example, when $N = 8$, $\lambda = 128$, the ciphertext occupies approximately 260 bits, and when $N = 64$, $\lambda = 128$, the ciphertext occupies approximately 288 bits.

## 6 Conclusion and future work

In this paper, we focus on the issue of the NS-SLA audit model of 5G and proposed a blockchain-based audit model which considers the privacy protection. In order to ensure the authenticity of the monitored data, both the customer and network operators are responsible for monitoring the service. Besides, these monitored parameters will be encrypted by the TORE, which ensures the security of data stored on the blockchain. All the audit tasks are implemented by the designed smart contract, which ensures the audit data over ciphertexts without leaking privacy and execute punishment for the offending party automatically. In the future work, we plan to further reduce the cost of the scheme.

### Abbreviations

NS-SLA: Service level agreement of network slicing; ETSI: European Telecommunications Standards Institute; NFV: Network function virtualization; GPP: 3rd Generation partnership project; QoS: Quality of service; TPM: Trusted platform module; TPA: Third-party auditors; OPE: Order-preserving encryption; ORE: Order-revealing encryption; TORE: A trapdoor order-revealing encryption; CSMF: Communication service management function; uRLLC: Ultra-reliability and low-latency communication; mMTC: Massive machine-type communication; eMBB: Enhanced mobile broadband; BS: Base station; MIMO: Multiple input and multiple output.

### Acknowledgements

We gratefully acknowledge the people who gave meticulous and valuable comments on this paper and the anonymous reviewers who spent the valuable time in reviewing our paper.

### Authors' contributions

KX and ZG proposed the main idea. ZG and YH are the main writers of this paper. GX, CW, and YT analyzed the results and discussed the function compare. YH is the corresponding author of this paper. All authors read and approved the final manuscript.

### Funding

This work was supported in part by Beijing Municipal Natural Science Foundation (M21029), in part by National Key R&D Program of China (2018YFB1800302), and in part by the Natural Science Foundation of China under Grant 61802005.

## Declarations

### Competing interests

The authors declare that they have no competing interests.

Received: 7 January 2021 Accepted: 2 August 2021
Published online: 11 August 2021

### References

1. E.H. Bouzidi, A. Outtagarts, A. Hebbar, R. Langar, R. Boutaba, Online based learning for predictive end-to-end network slicing in 5g networks, in *ICC 2020—2020 IEEE International Conference on Communications (ICC)* (2020), pp. 1–7. https://doi.org/10.1109/ICC40277.2020.9148926
2. M. Chahbar, G. Diaz, A. Dandoush, C. Cérin, K. Ghoumid, A comprehensive survey on the e2e 5g network slicing model. IEEE Trans. Netw. Serv. Manag. (2020). https://doi.org/10.1109/TNSM.2020.3044626
3. J. Cao, M. Ma, H. Li, R. Ma, Y. Sun, P. Yu, L. Xiong, A survey on security aspects for 3g pp 5g networks. IEEE Commun. Surv. Tutor. **22**(1), 170–195 (2020). https://doi.org/10.1109/COMST.2019.2951818
4. S. Narin, South Korea to Launch World's First National 5G Networks. https://www.voanews.com/silicon-valley-technology/south-korea-launch-worlds-first-national-5g-networks
5. L. Hardesty, The 5G of T-Mobile, Verizon and AT&T all rank badly for different reasons. https://www.fiercewireless.com/5g/5g-t-mobile-verizon-and-at-t-all-rank-badly-for-different-reasons
6. I. Fogg, Benchmarking the global 5G user experience—October update. https://www.opensignal.com/2020/10/13/benchmarking-the-global-5g-user-experience-october-update
7. D. Bega, M. Gramaglia, A. Banchs, V. Sciancalepore, K. Samdanis, X. Costa-Perez, Optimising 5g infrastructure markets: the business of network slicing, in *IEEE INFOCOM 2017—IEEE Conference on Computer Communications* (2017), pp. 1–9. https://doi.org/10.1109/INFOCOM.2017.8057045

8. S. Zhou, L. Wu, C. Jin, A privacy-based SLA violation detection model for the security of cloud computing. China Commun. **14**(9), 155–165 (2017)
9. M. Macías, J. Guitart, Analysis of a trust model for SLA negotiation and enforcement in cloud markets. Fut. Gener. Comput. Syst. **55**, 460–472 (2016)
10. F. Nawaz, O. Hussain, F.K. Hussain, N.K. Janjua, M. Saberi, E. Chang, Proactive management of SLA violations by capturing relevant external events in a cloud of things environment. Fut. Gener. Comput. Syst. **95**, 26–44 (2019)
11. S. Nakamoto, A. Bitcoin, A peer-to-peer electronic cash system. Bitcoin (2008).https://bitcoin.org/bitcoin.pdf
12. W. Wang, D.T. Hoang, P. Hu, Z. Xiong, D. Niyato, P. Wang, Y. Wen, D.I. Kim, A survey on consensus mechanisms and mining strategy management in blockchain networks. IEEE Access **7**, 22328–22370 (2019). https://doi.org/10.1109/ACCESS.2019.2896108
13. A blockchain based witness model for trustworthy cloud service level agreement enforcement, in *IEEE INFOCOM* (IEEE, 2019), pp. 1567–1575
14. A.T. Wonjiga, S. Peisert, L. Rilling, C. Morin, Blockchain as a trusted component in cloud SLA verification, in *ACM International Conference on Utility and Cloud Computing Companion* (2019), pp. 93–100
15. A. Papageorgiou, A. Fernández-Fernández, L. Ochoa-Aday, M.S. Peláez, M. Shuaib Siddiqui, SLA management procedures in 5g slicing-based systems. in *2020 European Conference on Networks and Communications (EuCNC)* (2020), pp. 7–11. https://doi.org/10.1109/EuCNC48522.2020.9200904
16. R. Agrawal, J. Kiernan, R. Srikant, Y. Xu, Order preserving encryption for numeric data, in: ACM SIGMOD (2004), pp. 563–574
17. N. Chenette, K. Lewi, S.A. Weis, D.J. Wu, Practical order-revealing encryption with limited leakage, in *International Conference on Fast Software Encryption* (Springer, 2016), pp. 474–493
18. K. Lewi, D.J. Wu, Order-revealing encryption: new constructions, applications, and lower bounds, in *ACM SIGSAC* (2016), pp. 1167–1178
19. N. Kaaniche, M. Mohamed, M. Laurent, H. Ludwig, Security SLA based monitoring in clouds. In: IEEE EDGE (IEEE, 2017), pp. 90–97
20. J. Kang, Z. Xiong, D. Niyato, D. Ye, D.I. Kim, J. Zhao, Toward secure blockchain-enabled internet of vehicles: optimizing consensus management using reputation and contract theory. IEEE Trans. Veh. Technol. **68**(3), 2906–2920 (2019). https://doi.org/10.1109/TVT.2019.2894944
21. H. Nakashima, M. Aoyama, An automation method of SLA contract of web apis and its platform based on blockchain concept, in *IEEE ICCC* (IEEE, 2017), pp. 32–39
22. S. Singh, I. Chana, R. Buyya, Star: SLA-aware autonomic management of cloud resources. IEEE Trans. Cloud Comput. **8**(4), 1040–1053 (2020). https://doi.org/10.1109/TCC.2017.2648788
23. M. Franceschetti, J. Eder, Checking temporal service level agreements for web service compositions with temporal parameters, in *2019 IEEE International Conference on Web Services (ICWS)* (2019, 2019), pp. 443–445. https://doi.org/10.1109/ICWS.2019.00080
24. C. Parada, J. Bonnet, E. Fotopoulou, A. Zafeiropoulos, E. Kapassa, M. Touloupou, D. Kyriazis, R. Vilalta, R. Muñoz, R. Casellas, R. Martínez, G. Xilouris, 5GTANGO: A Beyond-MANO Service Platform (2018)
25. M. Touloupou, E. Kapassa, C. Symvoulidis, P. Stavrianos, D. Kyriazis, An integrated SLA management framework in a 5g environment, in *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)* (2019), pp. 233–235. https://doi.org/10.1109/ICIN.2019.8685916
26. Y. Liu, J. Peng, J. Kang, A.M. Iliyasu, D. Niyato, A.A.A. El-Latif, A secure federated learning framework for 5g networks. IEEE Wirel. Commun. **27**(4), 24–31 (2020). https://doi.org/10.1109/MWC.01.1900525
27. J. Kang, Z. Xiong, C. Jiang, Y. Liu, C. Miao, Scalable and communication-efficient decentralized federated edge learning with multi-blockchain framework (2020)
28. M. Naveed, S. Kamara, C.V. Wright, Inference attacks on property-preserving encrypted databases, in *ACM SIGSAC* (2015), pp. 644–655
29. D. Boneh, K. Lewi, M. Raykova, A. Sahai, M. Zhandry, J. Zimmerman, Semantically secure order-revealing encryption: Multi-input functional encryption without obfuscation, in *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (Springer, 2015), pp. 563–594
30. H.T. Nguyen, W. Zhao, J. Yang, A trust and reputation model based on Bayesian network for web services, in *IEEE ICWS* (IEEE, 2010), pp. 251–258
31. K. Xiao, F. Wang, H. Rutagemwa, K. Michel, B. Rong, High-performance multicast services in 5g big data network with massive mimo, in *2017 IEEE International Conference on Communications (ICC)* (2017), pp. 1–6. https://doi.org/10.1109/ICC.2017.7996723

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.