IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 17 May 2022, accepted 13 June 2022, date of publication 4 July 2022, date of current version 11 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3188110

Secure Data Storage and Sharing Techniques for Data Protection in Cloud Environments: A Systematic Review, Analysis, and Future Directions

ISHU GUPTA[1], (Member, IEEE), ASHUTOSH KUMAR SINGH[2], (Senior Member, IEEE), CHUNG-NAN LEE[1], (Member, IEEE), AND RAJKUMAR BUYYA[3], (Fellow, IEEE)

[1]Cloud Computing Research Center, Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan
[2]Department of Computer Applications, National Institute of Technology Kurukshetra, Kurukshetra 136119, India
[3]Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Melbourne, VIC 3010, Australia

Corresponding author: Ishu Gupta (ishugupta23@gmail.com)

This work was supported in part by the National Sun Yat-sen University, Kaohsiung, Taiwan; and in part by the National Institute of Technology, Kurukshetra, India.

ABSTRACT A large number of researchers, academia, government sectors, and business enterprises are adopting the cloud environment due to the least upfront capital investment, maximum scalability, and several other features of it. Despite the multiple features supported by the cloud environment, it also suffers several challenges. Data protection is the primary concern in the area of information security and cloud computing. Numerous solutions have been developed to address this challenge. However, there is a lack of comprehensive analysis among the existing solutions and a necessity emerges to explore, classify, and analyze the significant existing work for investigating the applicability of these solutions to meet the requirements. This article presents a comparative and systematic study, and in-depth analysis of leading techniques for secure sharing and protecting the data in the cloud environment. The discussion about each dedicated technique includes: functioning for protecting the data, potential and revolutionary solutions in the domain, the core and adequate information including workflow, achievements, scope, gaps, future directions, etc. about each solution. Furthermore, a comprehensive and comparative analysis of the discussed techniques is presented. Afterward, the applicability of the techniques is discussed as per the requirements and the research gaps along with future directions are reported in the field. The authors believe that this article's contribution will operate as a catalyst for the potential researchers to carry out the research work in the area.

INDEX TERMS Cloud computing, data privacy and security, data protection, data storage, data sharing, IoT, machine learning, cryptography, watermarking, access control, differential privacy, probabilistic approaches.

I. INTRODUCTION

Data is acknowledged as the most vital asset of an organization because it defines the uniqueness of every enterprise. It is the main foundation of information, knowledge, and ultimately the wisdom for correct decisions and actions. It might be helping to cure a disease, boost a company's revenue, make a building more efficient or be responsible for achieving the targets, and improving the performance [1]. Furthermore, storage, analysis, and sharing of data are the essential services required by any organization to upgrade its performance [2]. However, with the explosive evolution of data, enormous pressure emerges on the enterprises for storing the voluminous data locally [3]. Also, it has become difficult to explore the data due to limited resources [4]. Most businesses have shifted to the cloud for these services

The associate editor coordinating the review of this manuscript and approving it for publication was Mansoor Ahmed.

due to its several advantages such as on-demand service, scalability, reliability, elasticity, measured services, disaster recovery, accessibility, and many others [5]. Cloud computing is a paradigm that enables huge memory space and massive computation capacity at a low cost. It allows users to obtain the intended services across multiple platforms irrespective of location and time and consequently conveys an extensive convenience to the cloud users [6]. By migrating the local data management system into cloud storage and using cloud-based services, users can accomplish cost savings and productivity enhancements to manage projects and establish collaborations [7]. Therefore, individuals and organizations are shifting increasingly to the cloud for their multiple services [8]. With the growing expansion of cloud computing technologies, it is not difficult to imagine that almost all the businesses will be switched to the cloud in the foreseeable future [9].

Despite the multiple features offered by cloud computing, it encounters several impediments that may obstruct its fast growth, if not tackled appropriately [10]. Consider a real implementation, where an enterprise permits its staff or departments to store and share the data through the cloud. By exploiting the cloud, the enterprise can be completely released from the burden of maintaining and storing the data locally [11], [12]. Nevertheless, it also endures various security threats, which are the leading concerns of cloud users [13]. Firstly, outsourcing the data to the cloud servers signifies that the data is out of the users' control resulting in discomfort to the users because the outsourced data may comprehend sensitive and valuable information. Secondly, data sharing is frequently put into operation in a hostile and open environment, and the cloud server turned out to be a target of attacks. In the worst condition, users' data may be revealed by the cloud server itself for illegal profit [14], [15]. Furthermore, the data need to be shared among distinct relevant stakeholders, for instance, business partners, employees, customers, etc., interior or exterior of the organization's premises for upgrading the performance of the business. However, the recipient party can maltreat this data and disclose it purposefully or inattentively to some unauthorized third party [16], [17].

Fig. 1 represents a sharing environment where the data owners need to share the organization's valuable data to the cloud platform due to the limited storage and computational capacity of the enterprises and the multiple benefits of clouds. Furthermore, the cloud data is shared with multiple users as per different requirements for its utility purpose. However, the recipient party may leak the data after obtaining it. The data can be leaked by the involved parties or may steal by the unauthorized party through illegal access. Data leakage or loss may induce a severe threat to the organization's confidentiality. It can diminish the value of shareholders, decline the firm's rank and status, and destruct the enterprise's goodwill and reputation [18]. As the data is an important asset of an organization, thus it is essential to keep this asset secure. There arises a necessity for

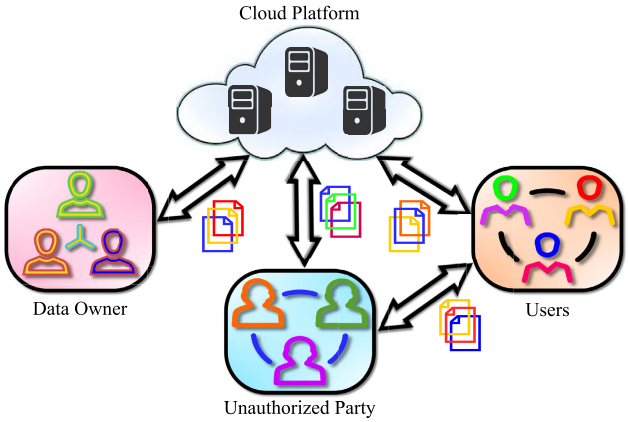


FIGURE 1. Block diagram of sharing environment.

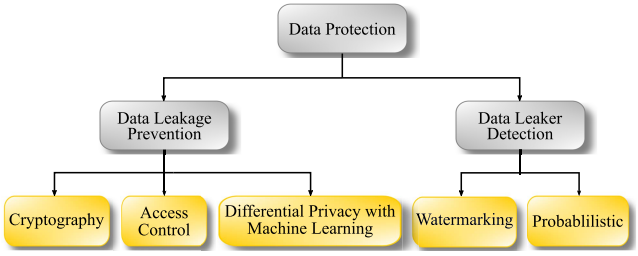


FIGURE 2. Major classification of data protection techniques.

solutions that can protect the data efficiently in the sharing environment.

A number of models for data protection in the cloud environment have been explored and developed for many applications. Typically, data protection is achieved through leakage prevention and leaker detection and this article concentrates on achieving efficient protection by preventing leakage and detecting the malicious entity responsible for leakage as depicted in Fig. 2. The major approaches for preventing data leakage are tailored by utilizing cryptography, access control mechanisms, and differential privacy with machine learning techniques while leaker detection is mainly achieved through watermarking and probabilistic techniques.

A. MOTIVATION

It was reported that 83% of the organizational workloads has shifted to the cloud platform by 2020 which raised to 90% within a year by 2021 [9], [19]. The cloud computing industry is forecast to rise with a 14.6% compound annual rate of growth to become a $300 billion industry by 2022 as of $188 billion in 2018 [20], [21]. Additionally, the connected IoT devices will reach 75 billion by 2025 which is 3 times the increment from 2019. IoT is the future and everything will continue to become more connected through technology that uses cloud services [22]. The data sharing and on-demand cloud access features of cloud computing have significantly reduced the data management cost while increasing the storage flexibility as

well as capacity [23], [24]. Despite that, it also sustains a crucial security threat to data confidentiality [25]. Precisely, the cloud users can not fully trust the Cloud Service Providers (CSPs) since the stored data files in the cloud may be confidential and sensitive [26]. Moreover, the data owners have serious concerns after sharing the data with the cloud due to the unavoidable loss of control over the data which clears the way for unauthorized data access [27]. Therefore, the security and privacy of sensitive data have become a major preoccupation for cloud users while using cloud computing services.

Also, the number of data leakage events together with the cost endured as a consequence of these leakages continuing to escalate is a serious matter of concern [28], [29]. According to Risk Based Security's (RBS) report, almost 22 billion data records have been disclosed within a single year 2021 surprisingly that is further expected to increase by 5% in 2022 [28]. The global average total cost of a data breach has reached $4.24 million in the year 2021 as per the IBM annual security report conducted by the Poneman institute which is the highest in the past 17 years [29]. Because of COVID-19, the average cost of a data breach is increased by $1.07 million due to remote work [22]. As a consequence, the data leakage problem is increasing day by day and it needs to be addressed. Thus, data protection has become a challenging task in the area of information security and cloud computing. There is a need for robust mechanisms that can address the existing problem effectively. The emerged challenge can be significantly overcome by preventing data leakage and recognizing the malicious entity that provokes data leakage. Several approaches have been discovered to protect the data in a cloud environment. Although a number of substantial solutions have been presented to mitigate the existing challenges in the domain, there arises a need to perform a systematic study of the existing solutions in order to find the applicability of these solutions as per the applications. Motivated by the significance and requirement for a better understanding of the current trends for sharing the data securely in the area of cloud computing, we present this analysis. For this purpose, a global level study is conducted and exhibited in this manuscript with the descriptions of the foremost techniques of data protection for the wide spectrum, easiness in obtaining related and eminent state-of-art existing solutions, their research gaps, future directions along with subsequent feasible solutions. The authors have first defined the general mechanism followed by provisioning in-depth detail and analysis of a particular technique with the aim of better understanding the concept and furnishing all the essential information conjointly for acquiring knowledge in the area. The relevant solutions of every individual technique, their merits, and scope are reported and further, analysis is performed to explore the relevancy of each technique as per the scenario. It is reckoned this article will contribute as a foundation for the emerging applications demanding data protection.

B. OUR CONTRIBUTION

The main contributions of the article are summarized as follows:

1) This work reviews the major and significant existing techniques for data protection through secure sharing in the cloud environment.
2) We provide the following-mentioned details about each of the technique (a) how it works for data protection and (b) the qualitative, outstanding, and primary solutions in the area. Furthermore, we present the potential and valuable information like the working, implementation environment, achievement, scope of the given model, etc., about each discussed solution in the tabular format to easily grab the core of the method along with its applications.
3) A comparative and comprehensive analysis of the discussed techniques are performed and exposed in a concise form. Furthermore, it is investigated which technique is best suited as per the requirements.

C. ORGANIZATION OF THE PAPER

Sections II to VI analyze the cryptography, access control, differential privacy with machine learning, watermarking, and probability techniques individually. Each section elaborates the following descriptions of the designated technique (A) the functioning of the technique with the help of a block diagram for protecting the data in the cloud environment (B) the remarkable contribution which is relevant and justifiable to identify the work done and the research gaps in the domain (C) the core information about every described solution is summarized and presented in a tabulated form for the ease of grabbing the necessary and sufficient details to carry out the further work. Section VII accomplishes a comparative analysis among the discussed technique and exposes the optimality of techniques as per the circumstances. Finally, the conclusion of the analysis performed and future remarks are reported in section VIII.

II. CRYPTOGRAPHY BASED MODELS

Let E_τ is the set of entities to be encrypted, $\mathcal{S}_\mathcal{K}$, $\mathcal{PB}_\mathcal{K}$, and $\mathcal{PV}_\mathcal{K}$ are the sets of secret, public, and private keys for encryption and decryption then the symmetric cryptography technique maps $\Phi_e : E_\tau \times \mathcal{S}_\mathcal{K} \rightarrow E_\tau^*$ and $\Psi_d : E_\tau^* \times \mathcal{S}_\mathcal{K} \rightarrow E_\tau$ such that $\Psi_d \Phi_e(\mathcal{E}_\tau, S_k) = \mathcal{E}_\tau$ and the asymmetric cryptography technique maps $\Phi_e : E_\tau \times \mathcal{PB}_\mathcal{K} \rightarrow E_\tau^*$ and $\Psi_d : E_\tau^* \times \mathcal{PV}_\mathcal{K} \rightarrow E_\tau$ such that $\Psi_d \Phi_e(\mathcal{E}_\tau, PB_k) = \mathcal{E}_\tau \; \forall \; \mathcal{E}_\tau \in E_\tau$, $S_k \in \mathcal{S}_\mathcal{K}$, $PB_k \in \mathcal{PB}_\mathcal{K}$, $PV_k \in \mathcal{PV}_\mathcal{K}$ where $\mathcal{E}_\tau^* \in E_\tau^*$ is the set of encrypted documents.

The symmetric cryptography technique $(E_\tau, E_\tau^*, \mathcal{S}_\mathcal{K}, \Phi_e, \Psi_d)$ consists of three functions is defined as-

- The key generator function $K_{gen}(CG)$ as shown in Eq. (1) generates a key S_k for the given security factor $\mathcal{S}^\mathcal{F}$.

$$S_k = K_{gen}(CG) \; \forall \; S_k \in \mathcal{S}_\mathcal{K} \tag{1}$$

- The encryption function $\Phi_e: \mathcal{E}_\tau \times S_k \rightarrow \mathcal{E}_\tau^*$ takes the original entity $\mathcal{E}_\tau$ and the key S_k as an input and generates an encrypted entity $\mathcal{E}_\tau^*$ as given in Eq. (2).

$$\mathcal{E}_\tau^* = \Phi_e(\mathcal{E}_\tau, S_k) \; \forall \; \mathcal{E}_\tau \in E_\tau \wedge S_k \in \mathcal{S}_\mathcal{K} \wedge \mathcal{E}_\tau^* \in E_\tau^* \tag{2}$$

- The decryption function $\Psi_d: \mathcal{E}_\tau^* \times S_k \rightarrow \mathcal{E}_\tau$ generates the original entity $\mathcal{E}_\tau$ as output by considering the conceivably encrypted entity $\mathcal{E}_\tau^*$ and the key S_k as an input as depicted in Eq. (3).

$$\mathcal{E}_\tau = \Psi_d(\mathcal{E}_\tau^*, S_k) \; \forall \; \mathcal{E}_\tau^* \in E_\tau^* \wedge S_k \in \mathcal{S}_\mathcal{K} \wedge \mathcal{E}_\tau \in E_\tau \tag{3}$$

The asymmetric cryptography technique $(E_\tau, E_\tau^*, \mathcal{PB}_\mathcal{K}, \mathcal{PV}_\mathcal{K}, \Phi_e, \Psi_d)$ consists of three functions is defined as-

- The key generator function $K_{gen}(CG)$ given in Eq. (4) generates the keys PB_k and PV_k for the given security factor $\mathcal{S}^\mathcal{F}$.

$$PB_k, PV_k = K_{gen}(CG) \; \forall \; PB_k \in \mathcal{PB}_\mathcal{K} \wedge PV_k \in \mathcal{PV}_\mathcal{K} \tag{4}$$

- The encryption function $\Phi_e: \mathcal{E}_\tau \times PB_k \rightarrow \mathcal{E}_\tau^*$ takes the original entity $\mathcal{E}_\tau$ and the key PB_k as an input and generates an encrypted entity $\mathcal{E}_\tau^*$ as depicted in Eq. (5).

$$\mathcal{E}_\tau^* = \Phi_e(\mathcal{E}_\tau, PB_k) \; \forall \; \mathcal{E}_\tau \in E_\tau \wedge PB_k \in \mathcal{PB}_\mathcal{K} \wedge \mathcal{E}_\tau^* \in E_\tau^* \tag{5}$$

- The decryption function $\Psi_d: \mathcal{E}_\tau^* \times PV_k \rightarrow \mathcal{E}_\tau$ generates the original entity $\mathcal{E}_\tau$ as output by considering the conceivably encrypted entity $\mathcal{E}_\tau^*$ and the key PV_k as an input as shown in Eq. (6).

$$\mathcal{E}_\tau = \Psi_d(\mathcal{E}_\tau^*, PV_k) \; \forall \; \mathcal{E}_\tau^* \in E_\tau^* \wedge PV_k \in \mathcal{PV}_\mathcal{K} \wedge \mathcal{E}_\tau \in E_\tau \tag{6}$$

The building block of the cryptography technique is demonstrated in Fig. 3. Documents $\mathbb{D} = \{D_1, D_2, \ldots, D_n\}$ are secured by encrypting it with the help of individual keys $\mathbb{K} = \{K_1, K_2, \ldots, K_n\}$ relatively and generated encrypted documents $\mathbb{D}^\mathbb{E} = \{D_1^E, D_2^E, \ldots, D_n^E\}$ are passed to multiple stakeholders. The receiving party decrypts the acquired encrypted documents $\mathbb{D}^\mathbb{E} = \{D_1^E, D_2^E, \ldots, D_n^E\}$ by using the shared keys $\mathbb{K} = \{K_1, K_2, \ldots, K_n\}$ and obtains the plain documents $\mathbb{D} = \{D_1, D_2, \ldots, D_n\}$. These documents are used by the receiving party after decrypting it.

Kao *et al.* [30] presented a user-centric key management scheme named uCloud to protect the cloud. In uCloud, the data of users is indirectly encrypted through RSA by utilizing users' public keys. The users' private keys are stored on the users' mobile devices instead of users' PCs or servers. Furthermore, the two-dimensional (2D) barcode images are exploited to express the users' private keys which are further employed for the decryption of users' sensitive

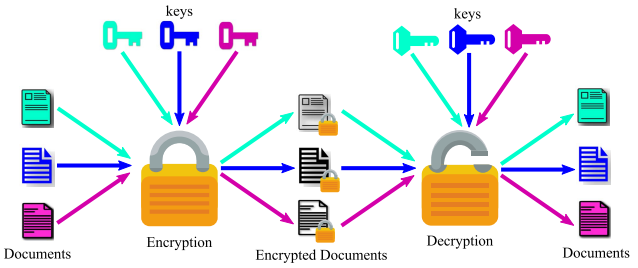


FIGURE 3. Birds-eye view of cryptography based models.

data. Al-Haj *et al.* [31] provided the two crypto-based algorithms to provide confidentiality, integrity, and authenticity to the data. They introduced a cryptographic function by using the hash code and symmetric keys to protect the data. The integrity and authenticity are provisioned by applying the elliptic curve digital signature algorithm. Additionally, the advanced encryption standard-Galois counter mode is used with the whirlpool hash function to support authenticity and confidentiality.

Liang *et al.* suggested a Ciphertext-Policy Attribute-Based Proxy Re-Encryption Scheme for the secure sharing of cloud data [32]. An enhancement of re-encryption and re-encryption key generation phases is introduced which minimized the communication and computational cost. A data owner is authorized in the scheme to assign the access rights of the encrypted data stored on a cloud system to others. A file hierarchy attribute-based encryption scheme is proposed by Wang *et al.* in [15] for securing the data in the cloud environment. This scheme used an access structure layered model to unravel the issue of sharing various hierarchical files and also demonstrated the protection of the file hierarchy-ciphertext policy-attribute based encryption (FH-CP-ABE) scheme which can effectively hinder the chosen plaintext attacks (CPA) under the assumption of Decisional Bilinear Diffie-Hellman (DBDH). The results showed that the cost of storage and complexity of computation is less in terms of encryption and decryption as compared to CP-ABE. The disadvantage of this scheme is that the computation cost is increased dynamically when the common attributes and an integrated ciphertext are desirable to be computed only once by the data owner.

Liu *et al.* [33] proposed a fair data access control scheme for cloud storage. In the scheme, a fair key reconstruction is performed to resist the access of shared data and none of the users exchanged their shares. A large number of fake keys are generated in the proposed scheme for obfuscating the decryption key of the shared data. Theoretical analysis of this scheme showed that all the shares are always contributed by their corresponding users which enables them to reconstruct the fair decryption key each time. Moreover, the performance evaluation demonstrated that the computation delay and communication costs are reduced, but the authentication scheme was not efficient in the scheme. A CP-ABE scheme is proposed by Liu *et al.* in [34] to reduce the computation

cost of heavy decryption at the user end which increases with respect to the complexity of access policy. This system facilitated decryption outsourcing, revocation attributes, and policy updating while attributes of the user are changed. The rigorous tests are implemented to analyze the performance of the proposed scheme which is measured in terms of storage overhead and processing power, however, it lacks in terms of privacy protection.

For mobile cloud computing, a lightweight data sharing scheme (LDSS) is proposed by Li *et al.* [6]. LDSS enhanced the structure of the access control tree by adopting the CP-ABE scheme to stimulate the mechanism applicable for mobile cloud environments. A large portion of the computation is displaced to external proxy servers from mobile devices in this scheme. The overhead on the side of the mobile device is reduced in LDSS when the data is shared by the users in the mobile cloud environments. Zaghloul *et al.* proposed a Privilege-based Multilevel Organizational Data-sharing (P-MOD) scheme in [2]. In P-MOD, the attribute-based encryption mechanism is strengthened by incorporating a privilege-based access structure into it to operate the sharing and management of big data sets effectively. It is demonstrated by the experimental analysis that the P-MOD is more efficient in comparison to both CP-ABE [35] and FH-CP-ABE [15] schemes for a hierarchical organization with many levels to perform the encryption and decryption and generate the keys. Also, the cumulative total of operations is minimized in the P-MOD scheme compared to the hierarchical schemes HABE [36], [37] and FH-CP-ABE [15].

Li *et al.* [8] presented a Linear Secret Sharing Scheme (LSSS) matrix access structure based an effectual CP-ABE scheme to update the file dynamically and improve the efficiency of the policy in the cloud environment. The objective of the scheme is to resist the selected plaintext attacks (CPA), and reduce the storage consumption of the proxy cloud service provider (PCSP), the communication expense, and the computing cost of the data owner. The theoretical analysis and experimental simulation of the proposed scheme showed that it has outperformed Policy Update CP-ABE [38] in terms of effective handling of the policy changes and file updates. To ensure the data confidentiality and protect the personal privacy of the user, a privacy-preserving scheme of the hidden access policy CP-ABE (HP-CP-ABE) schemes with an efficient authority verification is proposed by Zhang *et al.* [13]. In this approach, an authority detection mechanism to verify the authorized user and complete the decryption process is designed. This scheme obtained a private key of the constant size which is independent of the number of user's attributes. Though transmission and storage costs are decreased by this approach, it is realized as a weak security model because it supports the AND policy only. A thumbnail of relevant models based on the cryptography technique comprising potential details is portrayed in Table 1.

III. ACCESS CONTROL BASED MODELS

The Access Control Mechanism *ACM* allows controlled exposure of the confidential data to the authorized entity based on data type, user type, user's privileges, and permissions. An Access Control Policy (*ACP*) is defined for data distribution among users. *ACP* consists of a tuple $(\mathbb{D}, \mathbb{U}, \mathbb{G})$ where $\mathbb{D}$ refers to a set of data objects $D_1, D_2, \ldots, D_n$ to be distributed, $\mathbb{U}$ denotes a set of users $U_1, U_2, \ldots, U_m$, and $\mathbb{G}$ is an expression or a set of expression that decide which D_i can be accessed by which U_j or which D_i can be allocated to which U_j or U_j is allowed to access which D_i. *ACP* can vary depending upon the situations and applications.

ACM provides the information flow control and is suitable for any organization if access rights and data classification are properly established. Without a proper definition of access rights, it cannot be decided whether or not the data $\mathbb{D}$ is being accessed by a legitimate U_j. It is important to be able to distinguish between $U_1, U_2, \ldots, U_m$ based on their type, privileges, and permissions for an effective *ACM*. There must be predefined user privileges and data secrecy levels to work properly. Access is normally granted to U_j with credentials that meet the organization's policy. Fig. 4 represents a conventional model for access control mechanism. Three users U_1, U_2, U_3 send the request through the internet for the six documents $D_1, D_2, \ldots, D_6$. An access control policy is applied based on the users attributes, data attributes, and other essential factors; and a subset of data for which the users U_j qualify is transferred among U_1, U_2, U_3 through the internet. Where U_1, U_2, U_3 receives the dataset $\{D_1, D_2, D_6\}$, $\{D_1, D_4, D_5\}$, and $\{D_3, D_4, D_5\}$ respectively.

Nabeel and Bertino proposed a privacy-preserving policy-based content sharing scheme in public clouds [54]. The approach utilized a privacy-preserving attribute-based key management scheme that protects the privacy of users while enforcing attribute-based ACPs. The data owner performs coarse-grained encryption, whereas the cloud performs fine-grained encryption on top of the owner encrypted data to minimize the overhead at the data owners while assuring data confidentiality from the cloud. For the dynamic members in the cloud, a secure data sharing scheme is presented in [27]. The users can securely obtain their private keys due to the verification of their public keys. Revoked users cannot get the original data even if they conspire with the untrusted cloud to secure the scheme against collusion attacks. Previous users have no need to update their private keys when a new user joins or a user is revoked from the group to support dynamic groups.

A threshold multi-authority CP-ABE access control scheme TMACS is provided in [23] for public cloud storage in which multiple authorities jointly manage a uniform attribute set. A combination of the traditional multi-authority scheme and TMACS scheme is employed to handle the attributes set as well as achieve security and system-level robustness in which attributes coming from different authority sets and multiple authorities in

TABLE 1. A capsulization of cryptography based models.

Model	Workflow	Implementation	Outcome	Drawbacks & Future Scope
A user-centric key management scheme (uCloud) for data protection in cloud environment Kao et al. (2013) [30]	• Used RSA encryption technique to encrypt the user data • Includes a hierarchical structure for data sharing and key backup	• The uCloud modules are implemented in C++, Java, and JSP, and deployed on the Hadoop platform with HBase DB • HTC G1 is utilized to deploy the mobile application	• Supports the following functionalities: user-centric data protection, flexible access control, data sharing, user-controlled decryption, private key storage on a mobile phone, and no access control matrix maintained by user unlike the schemes given in [39]–[42] which satisfy a proper subset only of these properties	• Trusted Platform Module (TPM) hardware [43]–[45] can be used to enhance the security level • Proxy re-encryption [46] and attribute-based encryption [47] can be combined together for fine-grained access control
Crypto-based algorithms Al-Haj et al. (2015) [31]	• For secured medical image transmission, two crypto-based algorithms are presented • The scheme employed the hash codes and strong cryptographic functions with internally generated symmetric keys	• 20 MRI DICOM brain images of size 256×256 pixels with a depth of 16 bits • The experiments are carried out on a Dell N5010 machine with MATLAB environment which is based on a graphical user interface (GUI)	• Encryption and decryption times for the second algorithm is lesser compared to first • Provided the authenticity, confidentiality, and integrity to both the pixel data and header of the DICOM images unlike the schemes [48] and [49] that achieved the aforementioned three properties for a proper subset of header and pixel data only	• The algorithms can be extended to tackle the multi-frame and multi-slice DICOM medical images • Tamper localization scheme can be incorporated for content-based integrity in place of the strict-integrity functionality
CP-ABPRE scheme Liang et al. (2015) [32]	• The scheme objective is to achieve adaptive chosen ciphertext secure security • Integrated the selective proof technique with dual system encryption technology	• Security analysis is provided • To prove the scheme, a proof framework which is introduced by Lewko and Waters [50] is followed	• The scheme is verified adaptively chosen-ciphertext secure • Applicable to many network applications	• The scheme is not able to detect the malicious entity in case of leakage • The scheme can be converted in the prime order bilinear group
FH-CP-ABE (File Hierarchy Attribute-Based Encryption) scheme Wang et al. (2016) [15]	• To address the problem of multiple hierarchical files sharing, a Layered model of access structure is presented • An integrated access structure is employed for the files encryption • The FH-CP-ABE scheme is capable of successfully resisting the chosen plaintext attacks (CPA) in accordance with the Decisional Bilinear Diffie-Hellman (DBDH) assumption which confirmed the security of the scheme	• Data Set: Random generated • Intel Core processor at 2.79 GHz and 1.96GB RAM operating Windows XP SP 3 • The Cpabe toolkit and Java Pairing-Based Cryptography library (JPBC) • Results are averages of 10 trials	• Reduced computational complexity in comparison with the CP-ABE scheme • Minimized the storage cost approximately equal to 69.6% and 81.3% as compared to the existing CP-ABE and FH-CP-ABE schemes respectively	• Computation cost is increasing dynamically in some cases

TABLE 1. ***(Continued.)*** **A capsulization of cryptography based models.**

A data access control scheme Liu et al. (2017) [33]	• A stimulus mechanism is devised to restrain the stinginess of rational users while exchanging the shares • To obfuscate the decryption key of the shared data, a wide range of fake keys are generated • All the users are encouraged in the scheme to access the shared data jointly and reconstruct the decryption key fairly	• Data set: Miracl library version 5.3.3 • 3.0 GHz Core 2 Duo CPU, Microsoft Windows 7–32 bits operating system, and 4 GB DDR3–1600 RAM • Implemented in C ++ • The simulation results are the mean of 1000 trials	• Communication cost and computation delay are limited on both the user and data owner • For number of shares $(n) = 20$, the average computation delay and communication cost of proposed scheme are 122.727 ms and 22.23 Kb respectively	• Authentica-tion scheme was not effective
A practical attribute-based encryption scheme Liu et al. (2018) [34]	• System model introduced the inclusion of a proxy server as a fresh entity • The major portion of the decryption workload is computed by the proxy server • To approve the attribute revocation, policy updating, and outsourcing decryption functions concurrently, a more vigorous CP-ABE scheme is proposed	• Standford Pairing-Based Crypto library • On a virtual machine platform: 10.0.2-build-1744117, equipped with a 3.20 GHz Intel Core CPU with 2.0 GB RAM with 32-bit Linux Ubuntu 12.04 • Results are the mean of 20 trials	• The computation cost heightens linearly relative to the number of distinct attributes between former and recent access policies • The time cost of directly re-encrypting the plaintext is almost consistent When 60% of the attributes between the two policies are different • Computation costs of the policies are nearly equal	• No privacy protection
A Lightweight Secure Data Sharing Scheme for Mobile Cloud Computing Li et al. (2018) [6]	• To ensure an efficacious access control over ciphertext, an LDSS-CP-ABE algorithm relying on Attribute-Based Encryption (ABE) is designed • Proxy servers are responsible to conduct the operations in ABE	• The experiments are performed on a Core 2 DUO machine having 2.0 GHz CPU running the Linux operating system • Used CPABE tools established by Bethencourt et al. [35]	• Ensured data privacy in mobile cloud • With the inclusion of a marginal additional cost on the server-side, LDSS reduced the computational overhead on the client-side • LDSS performed better than [51] When the number of revoked attributes grow faster	• When the key is accessed by the intruders, the privacy of the data is infringed • Data integrity and ciphertext retrieval can be incorporated with the scheme
Privilege-based Multilevel Organizational Data-sharing (P-MOD) scheme Zaghloul et al. (2019) [2]	• A data file is partitioned into multiple fragments based on data sensitivity and user privileges • Depending upon data user's privileges, each fragment of the data file is shared • Facilitates data sharing in hierarchical settings	• Real U.S. Census Income data set • Implemented in Java utilizing the Java Pairing-Based Cryptography (JPBC) library [52] and the CP-ABE toolkit [53] • Experiments are carried out on an Intel (R) Core (TM) i5-4200 M at 2.50 GHz and 4.00 GB RAM machine running the Windows 10 OS	• Secure against adaptively chosen-plaintext attack under the DBDH assumption • Better performance for storage space and computational complexity compared to the existing schemes, CPABE [35], HABE [36], [37], and FH-CP-ABE [15]	• Dealt with single untrusted entity only • The scheme can be utilized for smart contract development and future attribute-based secure data management

TABLE 1. *(Continued.)* **A capsulization of cryptography based models.**

Attribute-based encryption scheme with policy update and file update Li et al. (2019) [8]	• Proxy Cloud Service Provider (PCSP) commences the major work of the policy update • The security hazards that are resulting from the invariable secret value are avoided in the updated scheme • It resists Chosen Plaintext Attacks (CPA) under the decision q-parallel Bilinear Diffie–Hellman Exponent (BDHE) assumption	• Java Pairing-Based Cryptography (JPBC) library and cpabe toolkit • 160-bits elliptic curve group over 512-bits finite field • Used Java on the Windows 7 with Intel Core processor at 3.40 GHz and 4.00-GB RAM	• Policy and generated update key is around 0.005 ms • The cost of the update ciphertext is approximately 26 ms • Policy update as well as file update outperformed over PU-CP-ABE [38]	• Did not support the file update and policy update features • The file update and policy update can be solved with the help of blockchain technology
Hidden Access Policy CP-ABE (HP-CP-ABE) schemes Zhang et al. (2020) [13]	• An authority identification method is designed which avoid unnecessary computations of users in the decryption process • Achieved a private key of the constant size which is independent of the number of user's attributes • It diminished the storage and transmission costs	• Data Set: PBC library • 64-bit PC with Intel Core i5-6400 CPU at 2.70 GHz and 8 GB RAM	• The size of the secret key is consistent, although it expands linearly with respect to the number of attributes in the existing works • Size of ciphertext is much shorter compared to the state of the art schemes	• It supports AND policy only and results based on a weak security model

an authority-set jointly maintain a subset of the whole attribute set.

A hierarchical access control system is designed in [17] that provides inheritance of authorization to reduce the burden and risk in the case of a single authority. The scheme adopts CP-ABE with the constant-size ciphertext to solve the linear dependency of ciphertext size on the number of attributes and maintains the size of ciphertext and the computation of encryption and decryption at a constant value which reduces the extra overhead of space storage, data transmission, and computation. Ali *et al.* [55] proposed a security scheme for outsourced data to the cloud (DaSCE) that provides (a) key management (b) access control, and (c) file assured deletion. The scheme utilizes Shamir's threshold scheme to manage the keys. Access control is enforced to both data and key through the validity of policies and mutual authentication between the client and key managers, and client and cloud. Assured deletion is based on policies associated with the data file uploaded to the cloud.

Almutairi *et al.* [56] presented virtual resource management methodologies for a cloud environment by designing Role-Based Access Control (RBAC) policy that minimizes the threat of data exposure. The concept of sensitivity is utilized in multi-tenant data centers in terms of the degree of data sharing among tenants. Limited sharing implies a high sensitivity data center and high sharing of data means a low sensitivity data center. Xu *et al.* [26] proposed a fine-grained access control and data sharing scheme for dynamic user groups and on-demand services by 1) defining and enforcing access policies based on the data attributes; 2) permitting the key generation center to update user credentials, and 3) allowing computation tasks to be performed by untrusted CSPs without requiring any delegation key.

A time and attribute factors combined access control on time-sensitive data for public cloud storage (TAFC) method is proposed in [57] by embedding Timed-Release Encryption (TRE) into Ciphertext-Policy Attribute-based Encryption (CP-ABE). This scheme provides data owners with the capability to flexibly release the access privilege to different users at different times according to a well-defined access policy over attributes and release time. Table 2 outlines the considerable models relying on access control involving the vital descriptions.

IV. DIFFERENTIAL PRIVACY WITH MACHINE LEARNING BASED MODELS

A mechanism $\mathcal{M}_{\mathcal{N}}: \mathbb{D} \rightarrow Range(\mathcal{M}_{\mathcal{N}})$ satisfies ϵ-differential privacy if for any possible output $\mathcal{O}_{\mathcal{P}} \in Range(\mathcal{M}_{\mathcal{N}})$ and every pair $D_i, D_i' \in \mathbb{D}$ distinct in only one record as depicted in Eq. (7) where P_b denotes the probability and $\hat{e}$ signifies the exponent.

$$P_b[\mathcal{M}_{\mathcal{N}}(D_i) = \mathcal{O}_{\mathcal{P}}] \leq \hat{e}^{\epsilon} \cdot P_b[\mathcal{M}_{\mathcal{N}}(D_i') = \mathcal{O}_{\mathcal{P}}] \quad (7)$$

Differential privacy with machine learning aims to protect sensitive information by making the outputs of different queries differing in at most one record indistinguishable.

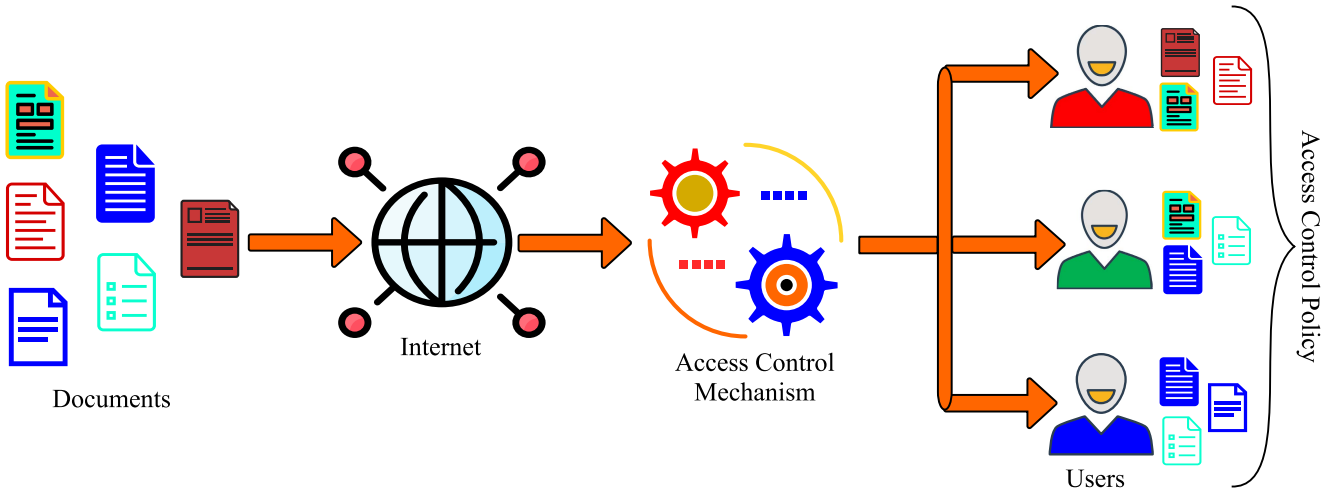


FIGURE 4. Schematic representation of access control based models.

ϵ-differential privacy is a popular approach to privacy protection for machine learning algorithms on data sets where $\epsilon > 0$ is a real number and predefined privacy parameter. It controls how much information is disclosed about an individual's data through statistical analysis and computation. The lesser the value of ϵ, the more powerful is privacy protection. The main idea of ϵ-Differential privacy in machine learning is to learn a simple rule automatically from the distributional information of the data set at hand without revealing too much about any single individual in the data set. Fig. 5 depicts a conventional example of privacy-preserving machine learning. The documents $D_1, D_2, \ldots, D_n$ of various types are protected through ϵ-differential privacy and made private followed by the machine learning to classify $D_1, D_2, \ldots, D_n$. The ϵ-differential privacy is applied over $D_1, D_2, \ldots, D_n$ where the statistical noises are embedded with the documents for preserving their privacy. Afterward, computation is performed over $D_1, D_2, \ldots, D_n$ through machine learning that classify these documents in their appropriate categories $\{A, B, C, D\}$.

Let E_τ is the set of entities to be applied differential privacy, $\mathcal{N}_\mathcal{G} \in \mathbb{R}$ is the set of generated noise that has to be embedded within the documents then the differential privacy technique maps $\Phi_e^*\colon E_\tau \times \mathcal{N}_\mathcal{G} \rightarrow E_\tau^*$ and $\Psi_d^*\colon E_\tau^* \rightarrow \mathcal{N}_\mathcal{G}$ such that $\Psi_d^*\Phi_e^*(\mathcal{E}_\tau, N_G) = N_G \,\forall\, \mathcal{E}_\tau \in E_\tau$ and $\forall\, N_G \in \mathcal{N}_\mathcal{G}$. The differential privacy technique can be represented as a tuple $(E_\tau, E_\tau^*, \mathcal{N}_\mathcal{G})$ comprises of three functions that are delineated as-

- The noise generator function $\mathcal{N}_{gen}(DP)$ generates noise N_G for the given security factor $\mathcal{S}^\mathcal{F}$ as shown in Eq. (8).

$$N_G = \mathcal{N}_{gen}(DP) \,\forall\, N_G \in \mathcal{N}_\mathcal{G} \tag{8}$$

- The noise embedding function $\Phi_e^*\colon \mathcal{E}_\tau \times N_G \rightarrow \mathcal{E}_\tau^*$ takes the original entity $\mathcal{E}_\tau$ and the generated noise N_G as an input and generates a noised entity $\mathcal{E}_\tau^*$ as depicted in Eq. (9).

$$\mathcal{E}_\tau^* = \Phi_e^*(\mathcal{E}_\tau, N_G) \,\forall\, \mathcal{E}_\tau \in E_\tau \wedge N_G \in \mathcal{N}_\mathcal{G} \wedge \mathcal{E}_\tau^* \in E_\tau^* \tag{9}$$

- The noise extraction function $\Psi_d^*\colon \mathcal{E}_\tau^* \rightarrow N_G$ extracts the embedded noise N_G as an outcome by exploring the conceivably noised entity $\mathcal{E}_\tau^*$ as an input given in Eq. (10).

$$N_G = \Psi_d^*(\mathcal{E}_\tau^*) \,\forall\, \mathcal{E}_\tau^* \in E_\tau^* \wedge N_G \in \mathcal{N}_\mathcal{G} \tag{10}$$

Yonetani *et al.* developed a Doubly Permuted Homomorphic Encryption (DPHE) based privacy-preserving mechanism [65] that enabled the multi-party protected scalar product and reduced the high computational cost. The experimental evaluation proved that the envisioned method is capable of achieving better performance in comparison with the state-of-the-art visual recognition approaches. The major disadvantage of DPHE is that at an instant, it supported one operation only i.e. either multiplication or addition. Hesamifard *et al.* [66] proposed a framework named CryptoDL in which remedies are provided for employing deep neural network algorithms over encrypted data. They developed a theoretical basis for the implementation of deep neural network algorithms in the encrypted domain. Additionally, a neural network technique is established within the practical limitations of current homomorphic encryption schemes. Although the scheme operates adequately for securing private data, the attention is not drawn to the requirement of protecting private data through multiple keys from individual data owners.

Li *et al.* [67] introduced a privacy-conserving outsourced classification in cloud computing (POCC) framework under various public keys. To assure the confidentiality of sensitive data without leakage, they applied a fully homomorphic encryption proxy technique. But the data owner and

TABLE 2. A Capsulization of access control based models.

Model	Workflow	Implementation	Outcome	Drawbacks & Future Scope
A privacy-preserving fine-grained delegated access control approach Nabeel et al. (2014) [54]	• To support expressive Access Control Policies (ACPs), a group key management scheme is utilized • Decomposed the ACPs and performed the two layers of encryption to relieve the overhead at the owner	• Experiments are conducted on a machine with an Intel Core 2 Duo CPU T9300 2.50 GHz and 4 GB memory running GNU/Linux kernel version 2.6.32 • Implemented in C/C++ • Utilized SHA-1, AES-256, and Adjacency list representation	• Privacy of users is preserved and the confidentiality of the data is assured from the cloud while delegating vast majority the access control enforcement to the cloud	• This scheme is not secure due to the weak protection of commitment in the identity token issuance phase [27] • The computational cost can be reduced by exploiting partial relationships among ACPs.
An anti-collusion data sharing scheme Zhu et al. (2016) [27]	• Keys are distributed in a secure way without any secure communication channels • A polynomial function is exploited to accomplish the secure user revocation	• Group managers and group members processes are implemented on a laptop with Core 2 T5800 2.0 GHz, DDR2 800 2 G, Ubuntu 12.04 X86 • Cloud process is executed on a laptop with Core i7-3630 2.4 GHz, DDR3 1600 8 G, Ubuntu 12.04 X64	• The scheme provides secure key distribution, access control, secure user revocation, data confidentiality, and protection from anti-collusion attack	• The scheme does not support the multiple untrusted parties • Usage of resources drops down
Threshold multi-authority CP-ABE access control scheme (TMACS) Li et al. (2016) [23]	• Utilized the combination of multi-authority CP-ABE and threshold secret sharing schemes to reduce the overhead of managing all attributes on a single authority • The master key is shared among the involved authorities, and an authorized user is capable of generating its secret key through the interaction with any t authorities	• Threshold value (t)-$3, 5, 10, 15$ • Number of attribute authorities (n)-$5, 10, 15, 20$ • Computation, communication, and storage overhead are computed to evaluate the performance of the TMACS	• The system is secure with a probability approaching 1 when the value of t is 5 rather than a larger value with 10 authorities • No additional storage and computation overhead and larger communication overhead in TMACS compared to Waters's scheme [58]	• One authority can leak the data of others using the shared master key • The method does not provide security when t or more authorities are compromised • The mechanism should be given for selecting the values of (t, n) and interaction protocols should be designed
A hierarchical attribute-based access control scheme Teng et al. (2017) [17]	• To reduce the overhead and threat of a single authority scenario, the scheme adopted a hierarchical authorization structure • The introduced structure maintained the number of bilinear pairing evaluations and the length of ciphertext to a constant • The fine-grained, flexible, and scalable access control of outsourced data in cloud computing is realized in the scheme	• Conducted on a machine running Windows O.S with a CPU (two cores) of 2.00 GHz, ROM of 2 GB • Simulation is based on PBC-0.5.12 and GMP libraries • The cloud environment is established through the use of Hadoop-1.0.4 and VMware Workstation	• The performance of the scheme is better with reference to computation cost and memory requirement compared to other schemes [37], [59], [60] • Assured data confidentiality, fine-grained access control, and dynamic authorization	• Becomes unsuitable for practical implementation when replicas of the same attributes are administered by another domain authorities • In the instance of complex organizations with multiple domain authorities, it is a challenging task to synchronize the attribute administration

TABLE 2. ***(Continued.)*** **A Capsulization of access control based models.**

Data Security for Cloud Environment (DaSCE) scheme Ali et al. (2017) [55]	• To prevent the single point of failure in case of the cryptographic keys, the scheme involved the multiple key managers where each manager hosts one share of key • Diffie-Hellman and digital signatures are employed in the scheme to attain mutual authentication among the involved parties • Message authentication code (MAC) and symmetric key are exploited in this scheme to ensure the data integrity	• Used C for implementation • The .Net cryptographic packages are utilized to perform the cryptographic operations • The files of 9 divergent sizes 0.3 KB, 1 KB, 10 KB, 30 KB, 50 KB, 100 KB, 500 KB, 1 MB, and 10 MB are taken into account to carry out the experiments • Z3 solver, SMT-Lib, and HLPN are used to analyze the working of the scheme	• Ensured data confidentiality • DaSCE provided high security and the keys are not compromised against a man-in-the-middle attack, however, the performance overheads are fewer in the FADE [61] scheme in contrast to the DaSCE scheme	• Does not protect the data fully from all the involved entities or unauthorized access • Does not provide the secure key distribution and user revocation • Secure data forwarding and group data sharing can be implemented by extending the methodology
Risk-aware virtual resource assignment mechanism Almutairi et al. (2018) [56]	• Role-Based Access Control (RBAC) policy is enforced • The notion of sensitivity is introduced for data sharing • Three heuristics (a) Single Move Heuristic (SMH) (b) Multi Move Heuristic (MMH) and (c) Best Fit Heuristic (BFH) are presented for virtual resource assignment and the relative performance is compared	• Used three types of cloud datacenters Low Sensitive Datacenter (LSD), Medium Sensitive Datacenter (MSD), and High Sensitive Datacenter (HSD) • Vary the percentage of the data center from 70 to 95 • The size of the problem is assumed 50 VMs and 70 roles at a minor scale while the problem size is supposed 200 VMs and 120 roles at a major scale. Additionally, 20 services are taken into consideration in both cases	• Partial Risk is higher in the case of LSD as compared to HSD • Attackability is greater in the case of HSD in contrast to LSD • The Relative Risk Error (RRE) is higher in HSD than LSD if heuristic does not contemplate some of the partitions • MMH and BFH are the preferred choices for a smaller and a larger problem size respectively	• Did not protect the data from the co-resident attack • Mechanism is not threat specific • Security attributes of the cloud such as the probability of an attack, the impact of each attack associated with the identified threat, and the client-specific security requirement can be taken into account
Revocable Attribute-Based Encryption (RABE) scheme Xu et al. (2018) [26]	• Combined techniques of ABE, identity-based encryption, ciphertext encoding mechanism, and subset-cover framework • To deal with the user revocation, the workload of the service provider is reduced	• Implemented in C with the PBC library • The experiments are conducted on a desktop with 3.40 GHz Intel (R) Core (TM) i5-3570 CPU and 8 GB memory running 64-bit Ubuntu 16.04 • Maximum number of attributes are 30 and number of users are 2^{30}	• More efficient and scalable than SSW [62] • The storage and computation cost are reduced by a coefficient of log T where T implies the lifetime of the bounded system	• The plaintext information of stored ciphertexts can be transmuted by the cloud server through assembling the invalidated credentials of revoked users which may rise to a critical security challenge [63]
Time and Attribute Factors Combined access control mechanism (TAFC) Hong et al. (2020) [57]	• To ensure the secure fine-grained access control for time-sensitive data, the CP-ABE and TRE schemes are integrated into public cloud storage • Distinct access policies for time-sensitive data according to the divergent access requirements are designed through the suitable placement of time trapdoors	• Performance is measured in terms of computation and communication cost of the data owner and central authority • Compared with related schemes, LoTAC [64] and a CP-ABE based approach named TasA in which the time is treated as an attribute	• Reduced the communication complexity of data owner and central authority's cost compared to LoTAC and TasA when the access control system accommodates a massive extent of shared data and a significant number of users • Protected the confidentiality of time-sensitive data	• There may emerge a single-point bottleneck situation for both performance and security due to the involvement of single central authority only for maintaining the entire attribute set • Supported single untrusted entity only

the storage servers are considered to lie in the equivalent trustworthy area despite the fact that the storage servers are completely trusted. However, in cloud computing, this assumption is no longer applicable because both the data owner and database servers are very likely to be within different domains. Li *et al.* [68] proposed a scheme for a classifier owner to delegate a remote server to provide the privacy-preserving classification service for users. They designed efficient classification protocols for two concrete classifiers i.e. Naive Bayes and hyperplane decision-based. The experiments were conducted on the LAN server over testing datasets from the UCI Machine Learning Repository. A drawback of this scheme is that it involves frequent interactions of the users while launching a classification query. Li *et al.* [69] proposed a Privacy-Preserving Machine Learning with Multiple Data Providers (PMLM) scheme to defend the privacy of the data sets. They used public-key encryption with a double decryption algorithm (DD-PKE) and ϵ-differential privacy to encrypt the data sets of different data providers and the cloud respectively. The experiments are conducted under diverse classical machine learning algorithms to show the performance of the protocol. However, the computational cost is high in the proposed solution as a consequence of its dependency upon integer factorization.

A scheme is developed by Gao *et al.* in [70] to prevent information disclosure against the substitution-then-comparison (STC) attack. They adopted a double-blinding strategy and designed a functional privacy-preserving classification mechanism for the Naive Bayes classifier to protect data privacy. Most of the computations were performed offline phase in the server to reduce the overhead of online computation and communication. However, their approach has failed to achieve the discovery of truth that protects privacy. A data protection scheme for privacy-preserving Naive Bayes learning over data, contributed by multiple providers is proposed by Li *et al.* [71] which enabled the training of Naive Bayes classifier over the dataset, which is provided jointly by different data owners. The result of the training was achieved ϵ-differential privacy while the training will not break each owner's privacy. In this approach, collusions are allowed and adversaries had the ability to forge and manipulate the data.

Ma *et al.* [24] provided a Privacy-Preserving Deep Learning (PDLM) method for addressing the issue of training the model over the encrypted data under multiple keys. The proposed mechanism trains the model based on stochastic gradient descent (SGD) and performs the feed-forward and back-propagation procedure based on an efficient privacy-preserving calculation toolkit. This scheme reduced the overhead of the storage and computational complexity. The experimental evaluation showed that the classification model offered very little accuracy and high computation cost. A discussion incorporating the significant information about the expressive models referring to differential privacy with machine learning is presented in Table 3.

V. WATERMARKING BASED MODELS

Let E_τ is the set of entities that have to be watermarked, $\mathcal{W}_\mathcal{K}$ is the set of keys used for watermarking, and $\mathsf{W}_\mathcal{M}$ is the set of all feasible watermarks comprises the information that the owner wants to embed such that $\mathsf{W}_\mathcal{M} \subseteq \{0,1\}^+$ then the watermarking technique maps $\breve{\phi}_e \colon E_\tau \times \mathsf{W}_\mathcal{M} \rightarrow E_\tau$ and $\breve{\delta}_d \colon E_\tau \rightarrow \mathsf{W}_\mathcal{M}$ such that $\breve{\delta}_d \breve{\phi}_e(\mathcal{E}_\tau, \mathtt{W}_\mathtt{m}) = \mathtt{W}_\mathtt{m} \; \forall \, \mathcal{E}_\tau \in E_\tau$ and $\forall \, \mathtt{W}_\mathtt{m} \in \mathsf{W}_\mathcal{M}$.

The symmetric watermarking technique $(E_\tau, E_\tau^*, \mathcal{W}_\mathcal{K}, \mathsf{W}_\mathcal{M}, \mathsf{W}_\mathcal{M}^*, \breve{\phi}_e, \breve{\delta}_d)$ composed of three functions that are outlined as-

- The key generator function $K_{gen}(WM)$ generates a key W_k for the given security factor $\mathcal{S}^\mathcal{F}$ as shown in Eq. (11).

$$W_k = K_{gen}(WM) \; \forall \; W_k \in \mathcal{W}_\mathcal{K} \tag{11}$$

- The watermark embedding function $\breve{\phi}_e \colon \mathcal{E}_\tau \times W_k \times \mathtt{W}_\mathtt{m} \rightarrow \mathcal{E}_\tau^*$ takes the original entity $\mathcal{E}_\tau$, the key W_k, and the watermark $\mathtt{W}_\mathtt{m}$ as an input and generates a watermarked entity $\mathcal{E}_\tau^*$ as depicted in Eq. (12).

$$\mathcal{E}_\tau^* = \breve{\phi}_e(\mathcal{E}_\tau, W_k, \mathtt{W}_\mathtt{m}) \quad \forall \, \mathcal{E}_\tau \in E_\tau \wedge W_k \in \mathcal{W}_\mathcal{K} \wedge \mathtt{W}_\mathtt{m} \in \mathsf{W}_\mathcal{M} \wedge \mathcal{E}_\tau^* \in E_\tau^* \tag{12}$$

- The watermark detection function $\breve{\delta}_d \colon \mathcal{E}_\tau^* \times W_k \times \mathcal{E}_\tau \rightarrow \mathtt{W}_\mathtt{m}^*$ extracts the watermark $\mathtt{W}_\mathtt{m}^*$ as output by taking into account the conceivably watermarked entity $\mathcal{E}_\tau^*$, the key W_k, and the original entity $\mathcal{E}_\tau$ as an input given in Eq. (13).

$$\mathtt{W}_\mathtt{m}^* = \breve{\delta}_d(\mathcal{E}_\tau^*, W_k, \mathcal{E}_\tau) \quad \forall \, \mathcal{E}_\tau^* \in E_\tau^* \wedge W_k \in \mathcal{W}_\mathcal{K} \wedge \mathcal{E}_\tau \in E_\tau \wedge \mathtt{W}_\mathtt{m}^* \in \mathsf{W}_\mathcal{M}^* \tag{13}$$

Then we require a similarity function ξ depicted in Eq. (14) that takes the two objects $\mathtt{W}_\mathtt{m}$ and $\mathtt{W}_\mathtt{m}^*$ to be compared and returns $\diagup$ if the two objects are identified as similar and $\diagdown$ otherwise.

$$\xi(\mathtt{W}_\mathtt{m}, \mathtt{W}_\mathtt{m}^*) = \begin{cases} \diagup, & \mathtt{W}_\mathtt{m}, \mathtt{W}_\mathtt{m}^* \text{ are similar} \\ \diagdown, & \text{otherwise} \end{cases} \tag{14}$$

A robust watermarking should satisfy the following properties-

- **Imperceptibility:** $\mathcal{E}_\tau^* \leftarrow \breve{\phi}_e(\mathcal{E}_\tau, W_k, \mathtt{W}_\mathtt{m}) \Rightarrow \xi(\mathcal{E}_\tau, \mathcal{E}_\tau^*) = \diagup \, \forall \, \mathcal{E}_\tau \in E_\tau, \; \forall \, W_k \in \mathcal{W}_\mathcal{K}$ and $\forall \, \mathtt{W}_\mathtt{m} \in \mathsf{W}_\mathcal{M}$ which means the original entity $\mathcal{E}_\tau$ and the watermarked entity $\mathcal{E}_\tau^*$ are similar.
- **Effectiveness:** $\mathcal{E}_\tau^* \leftarrow \breve{\phi}_e(\mathcal{E}_\tau, W_k, \mathtt{W}_\mathtt{m}) \Rightarrow \breve{\delta}_d(\mathcal{E}_\tau^*, W_k, \mathcal{E}_\tau) = \mathtt{W}_\mathtt{m}^*$ such that $\xi(\mathtt{W}_\mathtt{m}, \mathtt{W}_\mathtt{m}^*) = \diagup \, \forall \, \mathcal{E}_\tau \in E_\tau, \; \forall \, W_k \in \mathcal{W}_\mathcal{K}$ and $\forall \, \mathtt{W}_\mathtt{m} \in \mathsf{W}_\mathcal{M}$. It specifies that if the watermark $\mathtt{W}_\mathtt{m}$ is embedded by applying the key W_k then an identical watermark should be detected by exploiting the same key W_k. In other words, we can say, embedded watermark $\mathtt{W}_\mathtt{m}$ and extracted watermark $\mathtt{W}_\mathtt{m}^*$ should be similar.

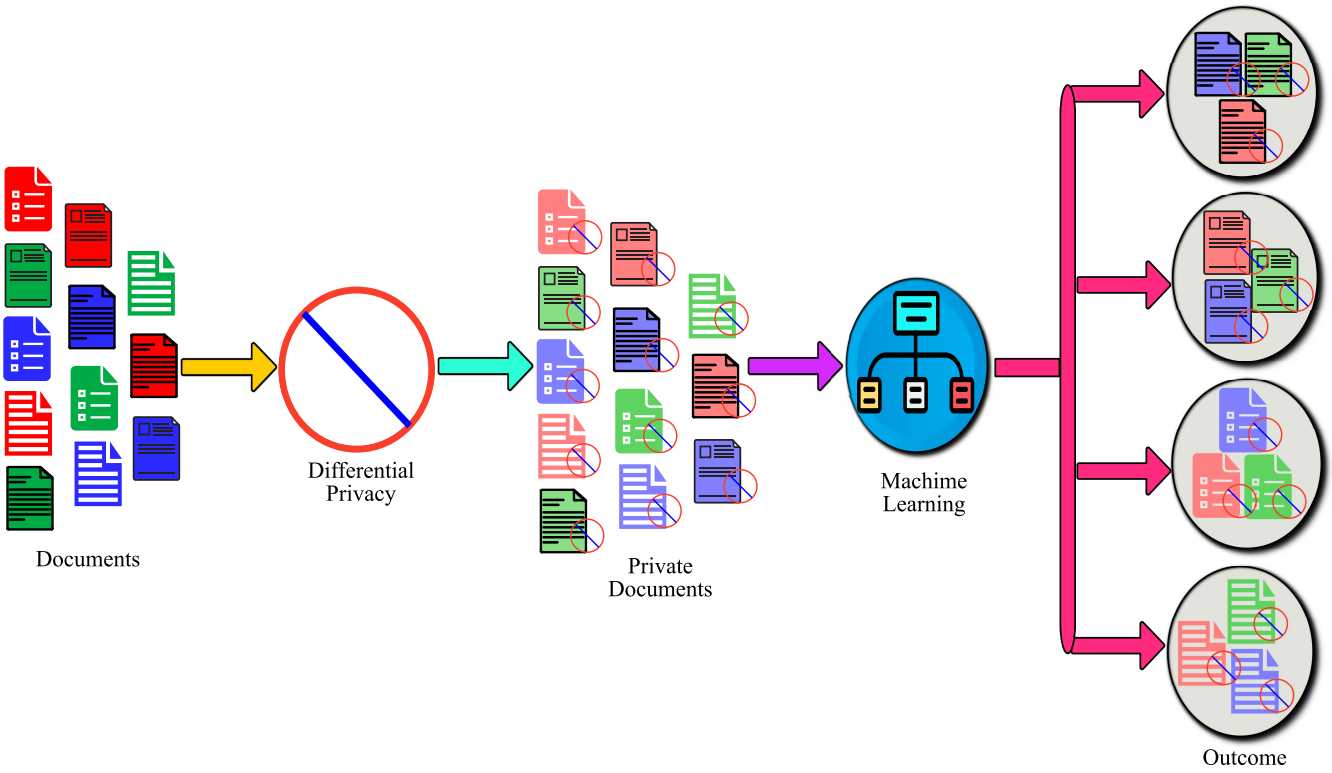


FIGURE 5. Standard model for differential privacy with machine learning.

- **Robustness:** For a watermark entity $\mathcal{E}_\tau^* = \breve{\phi}_e(\mathcal{E}_\tau, W_k, \mathbb{W}_\mathrm{m})$ where $\mathcal{E}_\tau \in E_\tau, W_k \in \mathcal{W}_\mathcal{K}$ and $\mathbb{W}_\mathrm{m} \in \mathsf{W}_\mathcal{M}$, there does not exist any polynomial time antagonist that can compute an $\mathcal{E}_\tau^{**} \in E_\tau$ given $\mathcal{E}_\tau^*$ and $\mathbb{W}_\mathrm{m}$ such that $\xi(\mathcal{E}_\tau^*, \mathcal{E}_\tau^{**}) = \nearrow$ and $\mathbb{W}_\mathrm{m}^{**} = \breve{\delta}_d(\mathcal{E}_\tau^{**}, W_k, \mathcal{E}_\tau)$ but $\xi(\mathbb{W}_\mathrm{m}, \mathbb{W}_\mathrm{m}^{**}) = \swarrow$ which means it should not be possible to change or remove the watermark $\mathbb{W}_\mathrm{m}$ by any antagonist effectively without cracking the similarity i.e without interpreting the entity inoperable.

Fig. 6 depicts the basic components involved in the process of watermarking technique. Data $\mathbb{D} = \{D_1, D_2, \ldots, D_n\}$ can be classified into various forms such as text, image, audio, video, relational, etc. A watermark $\mathbb{W}_\mathrm{m}$ is implanted in D_i using the watermark embedding process depending upon the category of D_i and the watermark $\mathbb{W}_\mathrm{m}^*$ is extracted using the watermark extraction process as per the category of D_i.

A technique is developed in [73] for fingerprinting relational data by extending the watermarking scheme given in [74]. A multi-bit watermark is combined with a collusion-resistant code. The arbitrary bit-string marks can be embedded in the relations as well as detected by the scheme. For the robustness properties of the scheme, the quantitative models are presented which demonstrated that the scheme is capable of detecting the embedded fingerprints against extensive kinds of attacks including collusion attacks. A scheme is provided for embedding the intangible watermark securely in the relational data via framing the watermarking in the form of a confined optimization case [75]. For this purpose, pattern search (PS) techniques in conjunction with genetic algorithms (GAs) are employed as well as data partitioning and threshold-based techniques are presented. The watermarks are embedded repeatedly and to enhance the watermark resilience, multiple attributes along with majority voting techniques were utilized for the watermark decoding phase. The performance evaluation showed that the technique is resilient to tuple insertion, alteration as well as deletion attacks, and watermark synchronization errors due to the employment of a partitioning approach where marker tuples are not demanded.

A mobile agent-based approach is developed in [76] for the identification of potential information leakage by automating the process of coloring and detecting the file systems of receptive hosts as well as monitoring the colored file systems. The detection capabilities are modularised and conditionally employed at the authority of a central control mechanism. The distributed reporting potential of mobile agent networks can perform future analyses of information leakage. Kumar *et al.* [77] introduced an approach based on watermarking that utilized the Bell-La Padula model for ensuring security via providing access control in the cloud environment. The approach embedded the client ID in the document whenever the cloud data is shared among the users. The guilty party is detected by extracting the embedded client ID from the discovered document. The model provides

TABLE 3. A capsulization of differential privacy with machine learning based models.

Model	Workflow	Implementation	Outcome	Drawbacks & Future Scope
Doubly Permuted Homomorphic Encryption (DPHE) Yonetani et al. (2017) [65]	• Utilized a homomorphic cryptosystem that can aggregate the local classifiers during encryption • Imposed sparsity constraints on local classifier updates and an encryption scheme named Doubly Permuted Homomorphic Encryption (DPHE) is proposed • Sparse data is decomposed into its constituent non-zero values and their relative support indices in DPHE	• Data Sets: Caltech 101, Caltech 256, CelebA and Life-logging • Single CPU of a MacBook Pro with Intel Core i7 at 2.9GHz • Gmpy2 and python-paillier	• Achieved better performance against state-of-the-art methods • 84% accuracy for face attribute recognition • 0.729 average Precision for sensitive place detection	• Either addition or multiplication operation is supported • Can be extended to learn significantly greater-dimensional models such as sparse convolutional neural networks
CryptoDL Framework Hesamifard et al. (2018) [66]	• Introduced techniques to approximate activation functions with polynomials of low degree • The approximated polynomials are utilized in neural networks and the performance of the new algorithms are analyzed • Developed a theoretical foundation and an approach based on Chebyshev polynomials is provided to generate the approximations	• Data Set: 17 data sets from the UC Irvine Machine Learning Repository • Neural Network Toolbox to implement the neural network • HELib for implementation • All the computations were performed on a virtual machine with 12 CPU cores, 48GB RAM, and Ubuntu 14.04	• Training over encrypted data is effective when batch learning is applied with satisfactory network performance • The framework achieved the training rate of 0.68 seconds/instance for 2 hidden layers, 6 features, and a batch size of 576 • Provides privacy-preserving and accurate classification and training • Outperforms state-of-the-art approaches based on secure multi-party computation and homomorphic encryption	• Didn't use any parallel programming techniques or graphics processing unit • Considered only continuous functions for approximation
Privacy-preserving Outsourced Classification in Cloud computing (POCC) Li et al. (2018) [67]	• Data providers, Evaluator, Clients and Crypto Service Provider are four entities considered in the system model • A proxy re-encryption fully homomorphic encryption technique is applied to preserve the privacy for data encryption • Allowed an evaluator to train the classification model with the help of the outsourced data of data providers • The classification model is stored in the encrypted form at the evaluator side	• Not Available	• A theoretical analysis is provided to prove the security of POCC	• No experimental demonstration • Data owners and servers are very rarely residing in different domains

TABLE 3. *(Continued.)* **A capsulization of differential privacy with machine learning based models.**

Outsourced privacy-preserving classification service over encrypted data Li et al. (2018) [68]	• A classifier owner outsources a classifier on a remote server after performing the encryption over it and a token is delivered afterward to authorized users • Classification conversation with the server can be initiated by any user by using the captured token • A query can be established by the authorized users from the remote server for the prediction of their instances in spite of the fact that the confidentiality is preserved about both the classifier and instance	• Data Sets: Splice-Junction Gene Sequences, Breast Cancer Wisconsin, SPECT Heart, and Balance Scale from the UCI Machine Learning Repository to train the Naive Bayes classifiers and test the analogous classification protocol • Cryptographic operations are implemented through GMP library • The prototype is executed in C++ on numerous different machines in the LAN that are associated with 1 Gbps Ethernet network	• Maximally a few seconds are taken by the classification protocols to run • The total time cost grows linearly with the increment in the number of users	• Interactions of the users are frequently involved when launching a classification query • For multi-user setting, outsourcing solutions with greater efficiency are requisite to be provided
Privacy-preserving machine learning with multiple data providers scheme Li et al. (2018) [69]	• To encrypt the data sets of multiple providers with different public keys, Public-Key Encryption in combination with a Double Decryption (DD-PKE) algorithm is used • In place of the encrypted data set, a randomized data set with ϵ-differential privacy is utilized to carry out the machine learning tasks • Through the use of ϵ-differential privacy, the noises are compounded according to the varying data analytics by the cloud server rather than the data providers	• Data Set: Krkopt, Glass, Cpu, Wine, and Abalone from the UCI Machine Learning Repository • The performance of DD-PKE cryptosystem is implemented on a PC with an Intel (R) Core (TM) i7-6500U CPU at 2.59 GHz and 8 GB RAM • The simulations of ϵ-differential privacy are executed on a PC with an AMD A4-3300M APU with Radeon (TM) HD Graphics 1.90 GHz and 6 GB RAM • The programs are developed in Java	• The efficiency of different classical machine learning algorithms is computed and shown using graphs • A theoretical security model is demonstrated to prove the scheme secure • Since the scheme transforms the encrypted data as it stands, due to this fact, the accuracy and efficiency of data processing are upgraded	• Experimen-tal analysis is very limited • The presented scheme is highly dependent on integer factorization which results in high computational cost
Privacy-preserving Naïve Bayes classifiers Gao et al. (2018) [70]	• Developed a double-blinding technique and collaborated this technique with oblivious transfer and additively homomorphic encryption techniques to hide the privacy of both the parties • In double-blinding, multiplicative and additive factor are two kinds of blinding factors • Classifier avoided the data exposure against the Substitution-Then-Comparison (STC) attack	• Data Set: Breast Cancer Wisconsin, Statlog Heart Dataset from the UCI Machine Learning Repository • The experiments are conducted on a PC with a 4-core 2.5 GHz Intel Core i5 CPU and 4 GB RAM • Implemented in C++ and compiled with g++ version 5.4.0 on a 64-bit version of Ubuntu (Ubuntu-16.04-desktop-amd64)	• Achieved accuracy up to 98.29% • Does not introduce more error rate in comparison with the original NB classifier	• The double blinding technique can be effectively integrated into other machine learning protocols

TABLE 3. *(Continued.)* **A capsulization of differential privacy with machine learning based models.**

Differentially private Naïve Bayes learning scheme Li et al. (2018) [71]	• In this scheme, the data is partitioned vertically and horizontally • Selected the Laplacian mechanism to preserve the differential privacy • Proposed algorithms do not involve heavy cryptographic tools • The designed aggregation method hides some statistics information	• Data Sets: the UCI Machine Learning Repository, Balance Scale, Breast Cancer Wisconsin, SPECT Heart, and Splice-Junction Gene Sequences for training Naive Bayes classifiers and testing the corresponding classification protocol • GMP library to implement cryptographic operations • The prototype is put into operation in C++ on a variety of dissimilar machines in the LAN that are linked to 1 Gbps Ethernet network	• The training result can achieve differential privacy while the training will not break the privacy of each owner • The scheme is practical for several applications	• Collusions are allowed among multiple entity or adversaries that had the ability to forge and manipulate the data
Privacy-preserving Deep Learning Model (PDLM) Ma et al. (2018) [24]	• The data of distinct data owners is encrypted by them using multiple keys and the encrypted data is outsourced to the service providers • The service provider transmits the training datasets to the untrusted cloud by encrypting it with the multiple keys of users for reducing the overhead on it • Adopt an effective privacy-preserving calculation toolkit • After acquiring the multi-key encrypted datasets, the cloud server converts these training datasets into encryptions through the product of all the participated public keys	• Dataset: MNIST which is formed of $60,000$ training handwritten digits and $10,000$ test ones from 0 to 9, CIFAR-10, composed of 32×32 color images in 10 classes • The practicality verification is performed on a machine with 2.4 GHz eight-core processor and 128 GB RAM • For the fabrication and training of LeNet deep learning, Torch7 nn [72] packages are exploited	• Training of deep learning model procures a sub-optimal output up to 5% loss in the perspective of classification accuracy • PDLM takes 2041.3 min and 2069.7 min to train the model in the case of CIFAR-10 and MNIST individually	• The classification accuracy is less with high computation cost • To accomplish the deep learning for the multiple data owners, a privacy-preserving mechanism with greater efficiency could be designed

security against the data leakage problem and is cost-effective in the context of space and time. However, the scheme is unproductive in an environment where the data objects are frequently accessed by multiple users. A technique that uses curvelet transforms is presented in [78] to hide patient information into their ECG signal. Curvelet transform decomposes the ECG signal into frequency sub-bands. A quantization approach is used to embed patient data into the coefficients whose values are around zero, in the high-frequency sub-bands. The experimental analysis proved that compared with the method which chooses random locations for the watermark, the proposed method performs better.

A generic framework called Lineage In Malicious Environment (LIME) based on data lineage is proposed by Backes *et al.* in [79] to protect the data in the vicious environment through the identification of the culprit entity. In this scheme, data is shared among multiple entities that can be either owners or consumers. To preserve the data in the malicious environment, a liable data transfer protocol is developed between the involved party via utilizing a robust combination of watermarking, signature primitives, and oblivious transfer techniques. This method considers the probable data leakage and the associated impediments at the design stage. The execution times are measured for distinct

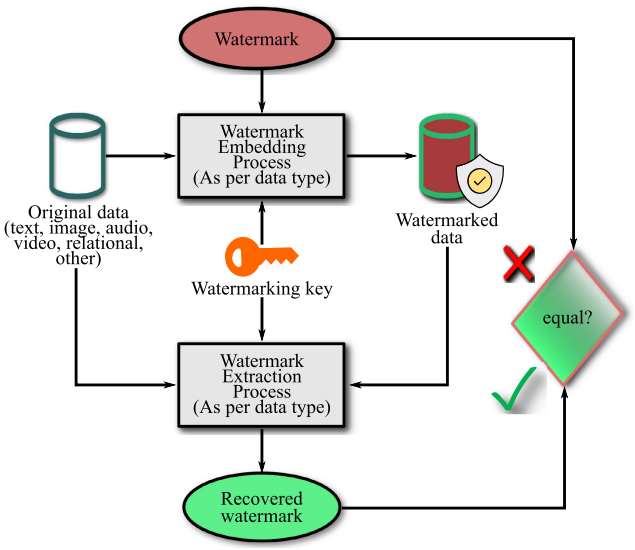


FIGURE 6. Key components of watermarking based models.

phases named watermarking, detection, oblivious transfer, encryption, and signature creation of the protocol. The framework is applied to the data leakage scenarios of social networks and data outsourcing. However, the proposed model cannot prevent the data from unauthorized access. The model can be extended to design the data leakage detection mechanisms for divergent scenarios and types of documents. The work provides future guidance in designing a verifiable lineage protocol for derived data.

A solution that uses role- and attribute-based access control for data exchange among services, including services hosted by untrusted environments is presented in [80] for privacy-preserving data exchange, data leakage detection, and prevention. The methodology employs Active Bundles (AB) that contain key-value pairs with values in encrypted form; metadata; access control policies and a policy enforcement engine. The active bundle mechanism provides data integrity and confidentiality and protects the data from malicious/curious cloud administrators. Implementation demonstrated that the data leakage detection mechanism imposes a 60.8% performance overhead. Amini *et al.* [81] proposed a statistical watermark detector for color images based on the Hidden Markov Model (HMM) to legitimate and secure online image transactions with a high detection rate. The HMM is used to trace the inter-channel dependencies among the contourlet coefficients of the color images. The superiority of the method against state-of-the-art methods [82]–[86] including Power-exponential [83], Cauchy [84], and Generalized Gaussian distributions [82] is confirmed by the experimental results.

An identity-based remote data integrity auditing approach is given by Shen *et al.* in [4] to conserve the integrity and protect the storage of sensitive information in the cloud. For this purpose, the integrity of files is verified through the use of signatures, and the cloud data is shared among multiple parties while hiding sensitive information. This method realized both the remote data integrity auditing and the files stored in the cloud are able to be shared and used by others on the condition that sensitive information is hidden in cloud storage.

A Genetic Algorithm and Histogram Shifting Watermarking (GAHSW) based reversible database watermarking technique is proposed in [87] to maximize the robustness and minimize distortion of the numerical relational database. In this approach watermarking is embedded by applying GA to select the best secret key for grouping the database. GAHSW causes less distortion and improves the robustness of watermarking as compared to state-of-the-art approaches in terms of robustness against malicious attacks and preservation of data quality. However, this approach is only applicable to a numerical database. A secure and robust digital text watermarking technique is proposed in [88] to provide copyright protection for text documents on local and cloud computing paradigms with the help of data mining techniques. This technique is applied to find suitable properties from the document for embedding the watermark. The proposed technique has attained a high level of imperceptibility where Peak Signal-to-Noise Ratio (PSNR) values are between 64.67% and 71.03%, and similarity (SIM) percentage is between 99.92% and 99.99%.

For the protection of medical images, Haddad *et al.* [89] presented a joint watermarking-encryption-compression (JWEC) scheme which has the ability to give access to watermarking-based security services from both encrypted and compressed image bit-streams. This scheme combines the bit-substitution watermarking with JPEG-LS and the AES block cipher algorithm in its cipher block chaining (CBC) mode, in a single operation performed on the entire image. The result demonstrates that watermark capacities are capable enough to support watermarking-based security services at the same time. A separable robust reversible watermarking in encrypted 2D vector graphics is proposed by Peng *et al.* [90] proposed to accomplish robust watermark extraction in plaintext as well as encrypted domain. In this scheme, a watermark mapping based on the polar coordinate system, hash-based message authentication code (HMAC), and erasure coding is built, which, achieves better invisibility and robustness against normal operations and malicious attacks compared with the existing methods. It is the first work reported on reversible watermarking in encrypted 2D vector graphics that can extract watermark in both domains but the attack to the reference vertex may fail the data extraction. Table 4 analyzes the remarkable models adapted from the watermarking technique coupled with imperative details.

VI. PROBABILITY BASED MODELS

The probability technique assesses the likelihood that an agent $U_j \in \mathbb{U}$ is accountable for exposing the given leaked data set $\mathbb{L}$ based on the overlap of his data with the leaked data and the data of other agents and based on the probability that objects can be guessed by other means.

Since the agents $U_1, U_2, \ldots, U_m$ have some of the data, it is reasonable to suspect them leaking the data. However, the agents can argue that they are innocent, and leaked data are obtained by the target through other means. For example, say that one of the objects in $\mathbb{L}$ represents a customer Z. Perhaps Z is also a customer of some other company, and that company provided the data to the target. Or perhaps Z can be reconstructed from various publicly available sources on the web. The more data in $\mathbb{L}$, the harder it is for the agents to argue they did not leak anything. Similarly, the rarer the objects, the harder it is to argue that the target obtained them through other means. For instance, if one of the $\mathbb{L}$ objects was only given to agent U_1, while the other objects were given to all agents, we may suspect U_1 more. To compute the probability, an estimate for the probability that values in $\mathbb{L}$ can be guessed by the target is required. For instance, say that some of the objects in $\mathbb{L}$ are e-mails of individuals. We can conduct an experiment and ask a person with approximately the expertise and resources of the target to find the e-mail of, say, 100 individuals. If this person can find, say, 90 e-mails, then we can reasonably guess that the probability of finding one e-mail is 0.9. On the other hand, if the objects in question are bank account numbers, the person may only discover, say, 20, leading to an estimate of 0.2. Data distribution strategies help in improving the probability of identifying a guilty user M_U. To identify a M_U with high confidence, it is needed to minimize $\frac{|\mathbb{W}_j^* \cap \mathbb{W}_k^*|}{|\mathbb{W}_j^*|} \forall j, k \in \{1, 2, \ldots, m\} \wedge j \neq k$. Therefore, the distribution strategies should distribute the data set $\mathbb{W}_1^*, \mathbb{W}_2^*, \ldots, \mathbb{W}_m^*$ with the objective given in Eqs. (15) and (16) to satisfy either one or both.

$$\underset{(over\ \mathbb{W}_1^*, \mathbb{W}_2^*, \ldots, \mathbb{W}_m^*)}{\text{minimize}} \quad \max_{\substack{j,k=1,2,\ldots,m \\ k \neq j}} \frac{|\mathbb{W}_j^* \bigcap \mathbb{W}_k^*|}{|\mathbb{W}_j^*|} \tag{15}$$

$$\underset{(over\ \mathbb{W}_1^*, \mathbb{W}_2^*, \ldots, \mathbb{W}_m^*)}{\text{minimize}} \quad \sum_{j=1}^{m} \frac{1}{|\mathbb{W}_j^*|} \sum_{\substack{k=1 \\ k \neq j}}^{m} |\mathbb{W}_j^* \bigcap \mathbb{W}_k^*| \tag{16}$$

Figs. 7a to 7c depict the three distribution strategies where four documents D_1, D_2, D_3, D_4 are distributed among four users U_1, U_2, U_3, U_4, each with a request R_1, R_2, R_3, R_4 of two documents. The third one avoids the full overlapping and is optimal among all three.

Papadimitriou and Garcia-Molina [103] proposed an agent guilt model based on a probabilistic approach to evaluate the likelihood of whether the data is revealed by one or more agents or it has been individually assembled by an unauthorized party through alternative means. This model assessed the maliciousness of various agents when the leaked data is found by the allocator at an illegal place. It is demonstrated that the judicious distribution of objects can assist in distinguishing the malicious entities with a remarkable distinction, especially, when the overlapping among the data acquired by the users is large.

Harel *et al.* [104] introduced a misuseability weight concept by delegating a score to the data sets as per their sensitivity. The scheme evaluated the sensitivity level of data which is revealed among the insiders, for diminishing the data misuse as well as data leakage incidents in the database system. Furthermore, the scheme estimated the insider's ability to exploit the sensitive data maliciously and also predicted the possibility of damage that could be resulting from the data leakage. A method is presented by Kumar *et al.* [105] to secure the data from unauthorized use. The allocation strategies are introduced that operate on account of no wait prototype and increase the chances of identifying the guilty party. The likelihood is assessed whether an individual agent was culpable for leaking a dataset or not.

For preventing the data leakage, a file distribution model is proposed by Fan *et al.* [106] which plans file allocation so as to minimize the overlapping between the received file sets of agents. Consequently, the model is proficient in discovering the origin of leakage with a large probability. The performance analysis revealed that the model is capable of detecting the sources of leakage as well as distinguishing the vicious agents efficiently. However, the achieved parameters are not compared with its baseline state-of-the-art distribution model [103]. For the textual data, a misuseability evaluator named TM-Score is defined in [107] which is an extension of the misuseability weight concept [104]. By utilizing the presented evaluator, the enterprises become capable of estimating the quantum of the detriment that is resulted from gradual and continuous exposure of textual content such as emails and documents caused by an insider. The degree of destruction is assessed by employing the quality, type, and amount of the revealed information. Sodagudi and Kurra provided a method in [108] to identify the malicious attackers in the mobile ad-hoc network (MANET) by considering the integration of routing protocol and cryptography technique. Allocation strategies are followed by the data distributor which results in less scope for data leakage to happen.

Guevara *et al.* presented an algorithm in [109] for data leakage detection by exploiting the property of anomalous user behavior. To accomplish the same, the user's operations are codified in a computer system by employing a dynamic structure, which permitted the extraction of a user's profile by following the sequences of actions from the historical database. The efficiency of the work is proved with reference to the low false-positive rate and high detection accuracy. However, its dependency on the historical data of the users for generating their behavioral patterns makes it a time-consuming process. Ezhilchelvan and Mitrani [110] have examined a system where several virtual machines shared a common physical machine and evaluated the probability of malicious co-residency in public clouds. The allocation of VMs to the physical machine is carried out by applying random and priority block policies while considering multiple security breaches simultaneously. The simulation results indicated the acceptable accuracy however, real-life experiments are required for confirming the accuracy of the reported method. An interpretation composing the essential

TABLE 4. A capuslization of watermarking based models.

Model	Workflow	Implementation	Outcome	Drawbacks & Future Scope
A symmetric marking scheme for relational databases Li et al. (2005) [73]	• The relational database is protected through the mark bit string that represents the individual buyers responsible for purchasing the database relation • The scheme utilized a single secret key to embed an arbitrary mark bit string into a relational data • The detection algorithm is used to test whether a relational data was marked with the help of a key and, in such case, it returns the original mark bit string that was embedded	• Forest cover type and real-life data sets were employed to implement the experiments • The data set has 61 attributes in addition to the id attribute as a primary key and 581, 012 tuples • For embedding the fingerprints, the foremost 10 integer-valued attributes are selected • The errors are computed to evaluate the results that are introduced through the insertion of fingerprints	• To evaluate the performance of the fingerprinting scheme, the two factors misdiagnosis false hit and misattribution false hit are calculated • A quantitative analysis is presented for all the assessment measures against proxy types of attacks	• Both the buyer and merchant possessed the copy of data with the same mark which made difficult to prove to a third party that the data was pirated and sold to a particular buyer • Asymmetric fingerprinting schemes can be designed on the basis of public-key cryptographic primitives in a two-party protocol
A resilient watermarking technique for relational data Shehab et al. (2008) [75]	• Performed secure embedding of a watermark in the relational data • Used pattern search techniques and genetic algorithms to optimize the watermarking of relational databases • To locate the partitions, a data partitioning technique independent of marker tuples is developed • Presented a threshold-based technique for watermark detection that reduces the probability of decoding error	• CPUs – 3.2 GHz, Intel Pentium IV, RAM – 512 MB, DB size – 5 MB, Tuples – 150000, Watermark – 16 bit, Number of partitions – 2048, Partition size – 10 • Normally and uniformly distributed synthetic data, real-life data consisting power consumption rates of customers for a year (synthetic & real-world data)	• Assessed computation times and represented a polynomial behavior w.r.t. data size • Supported the resiliency of the technique against tuple deletion, alteration & insertion attacks over [91] • Resilient to watermark synchronization errors compared to [91] • The probability of decoding errors is minimized over [91]	• Once the user is success in removing or altering the watermark then the method is not able to detect the malicious entity • The scheme is not capable of preventing the data leakage
An Information Leakage Detection (ILD) agent system Bishop et al. (2010) [76]	• The system automates the processes of converting a standard machine to a colored one • The system is able to add and modify detection capabilities while synchronously allowing conditional deployment of these capabilities • Dynamism is achieved without any or limited administrative interference due to the mobile agents	• Perl with Linux and XML is used to implement the mobile agent system • Image files and the algorithm from Peter Meerwald's watermarking library is used for watermarking • The performance is evaluated in deferred and real-time monitoring modes • A single host coupled with a local network of machines is exploited to evaluate the performance of the system	• Time to detect instances of leakage and to perform watermarking is measured • Number of file system events and CPU utilization with reference to differing scanning intervals were measured • Reduces the per-host administrative complexity • Reduced the overhead at host since the actions are performed by the agents themselves	• The system can successfully detect instances of information leakage, just in case, the agents are authentic and secure • The system cannot prevent data leakage • The future work leads to the inclusion of methods intended for detecting as well as blocking the covert channels before the information leakage occurrence

TABLE 4. *(Continued.)* **A capuslization of watermarking based models.**

A model for handling data leakage problem Kumar et al. (2014) [77]	• Protected the transmitted confidential information by recognizing the malicious entity who has leaked the organization's critical data • The proposed model detected the guilty user who is responsible for data leakage using watermarking, cryptography, and Bell-La Padula model	• Used AES-128 bits technique • Different size messages (Random) as a data set	• Performed encryption and decryption time comparison for DES, AES & RSA for various packet size • It can detect the exact malicious entity in real-time • The technique is worthwhile in a distributed computing environment for protecting the data against data leakage occurrence	• The leaker cannot be identified in case data is destroyed by the leaker agent • It is impracticable to expand the model for a web environment where the data objects are periodically accessed by enormous numbers of users
Curvelets-based ECG steganography technique Jero et al. (2016) [78]	• In ECG signals, the data of patients are hidden by the technique for data security • The watermark is embedded in the data by utilizing a quantization mechanism. For this purpose, the ones near zero are altered in ordered curvelet coefficients • The reversible watermarking process is executed to extract the data of the patients after transmission	• MIT-BIH database • Watermark imperceptibility and data loss are measured • Imperceptibility is measured using metrics mean square error (MSE), the Kullback-Leibler (KL) distance, percentage residual difference (PRD), and peak signal-to-noise ratio (PSNR) • Data loss is measured by computing BER	• The values of peak signal-to-noise ratio are large • No loss to the diagnosability information • The presented method is superior relative to the random locations method • The method can be employed for transferring the data of the patient safely	• The performance of the intended approach reduces with respect to the increment in the size of patient information • The scheme cannot protect the data when the malicious user successfully extracts the embedded watermark
LIME (Lineage in malicious environments) Backes et al. (2016) [79]	• Provided a framework for data flow across owner and consumer • For verifiably transferring the data among the involved parties, a liable data transfer protocol is presented • To cope with untrusted senders and receivers, a considerable integration of signature primitives, oblivious transfer, and watermarking techniques is employed in the scheme	• Lenovo ThinkPad model T430 with 4×2.6 GHz and 8 GB RAM • The protocol is implemented in C++, used pairing based cryptography (PBC) library [92] by utilizing GMP library [93] and BLS scheme [94] for oblivious transfer and signature primitives respectively, AES from Crypto++ [95] library for symmetric encryption, Cox algorithm [96] from Peter Meerwald's watermarking toolbox [97] for watermarking • Image of different size as a data set	• The information Leakage problem is addressed by employing lineage mechanism that demonstrably relating the malicious entity with the leakages • Computed the execution time for watermarking, oblivious transfer, signature primitives, encryption and detection	• The proposed framework cannot actively prevent the data leakages • For derived data, a verifiable lineage protocol can be designed as well as, for distinct scenarios and types of documents, data leakage detection methods can be developed as a future work

TABLE 4. ***(Continued.)*** **A capuslization of watermarking based models.**

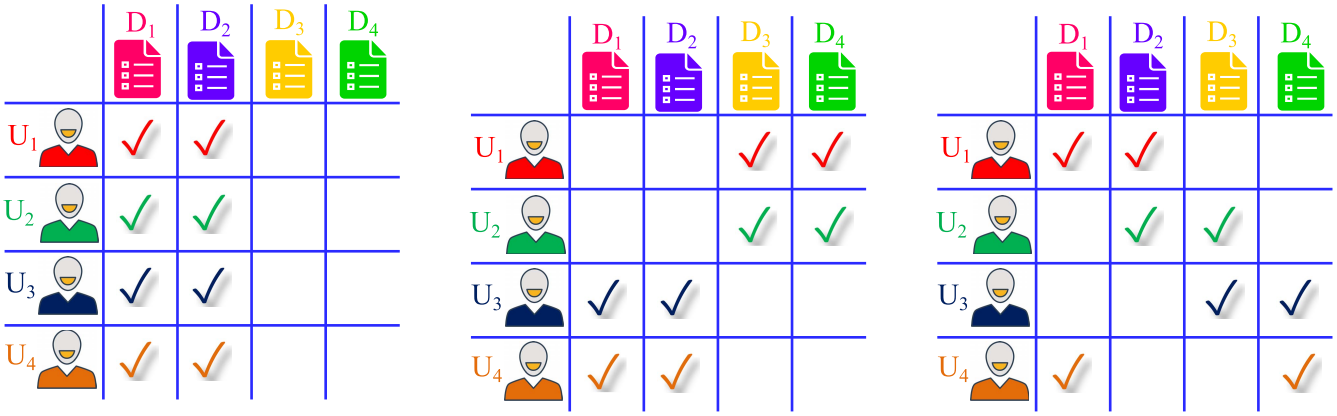
Damage assessment model for data leakages Ulybyshev et al. (2019) [80]	• Provided the solution for secure data exchange, data leakage prevention, and detection • The scheme utilized role & attribute-based access control, digital and visual watermarks, and symmetric encryption to address the data leakage problem • Supported fully decentralized architecture, where data is exchanged in a peer-to-peer network by the various services through forwarding the active bundles	• Apache bench utility and developer consoles [98] were used for RTT evaluation, Used AES for encryption • Handles non-relational databases	• Round trip time (RTT) for transmitting a data request as well as its retrieval from an active bundle is assessed and it has been reduced • Each of two decentralized and centralized data exchanges is supported by the scheme	• The scheme does not provide the protection against all the possible data leakages made by involved parties to the unauthorized entities
A channel-dependent statistical watermark detector Amini et al. (2019) [81]	• Proposed multichannel watermark detector utilizes the HMM for considering inter-scale dependencies among the sparse coefficients as well as the inter-channel dependencies among RGB channels of color images • The presented method performed the watermark detection in a sparse signal domain • A primitive binary hypothesis is exploited to accomplish the watermark detection	• Kodak dataset • The experiments are implemented on a personal computer with Intel Core i7 at 2.93-GHz and 8-GB RAM	• AUROC (Area under receiver operating characteristic): 0.9934 • High detection rates compared to the existing detector • The method detected the existence of watermark for the complicated instance, when quality factor $= 5$, and reported the relative improvement of 192.33%, 61.02%, 11.80%, 53.18%, and 5.29% over the methods in [82]–[86] respectively	–
An identity-based remote data integrity auditing and data sharing scheme Shen et al. (2019) [4]	• The scheme hid the sensitive information in the document for secure cloud storage • The data blocks related to the confidential information of the file are sanitized using a sanitizer and for the sanitized file, signatures of these data blocks are transformed into an authentic form • In the integrity auditing phase, the sanitized file's integrity is verified by utilizing the transformed signatures	• Implemented the experiments on a machine with an Intel Pentium 2.30GHz processor, 8GB RAM running Linux OS • Used C programming language with the GNU Multiple Precision Arithmetic (GMP) and Pairing-Based Cryptography (PBC) libraries • Set the length of a user identify field – 160 bits, the size of data file – 20MB consisting of $1,000,000$ blocks, the size of an element in the prime field – 160 bits, and the size of the base field – 512 bits	• The scheme satisfies the following properties: sensitive information hiding, data sharing, certificate management simplification, and public verifiability unlike the schemes given in [11], [99]–[102] which satisfy a proper subset only of these properties • The experimental results and the security analysis demonstrated that the desired efficiency and security is attained by the scheme	• The scheme is compared with [99] and it costs more computation overhead • The scheme can not prevent the data leakage • Data utility dropped down in this scheme

TABLE 4. *(Continued.)* **A capuslization of watermarking based models.**

Genetic Algorithm and Histogram Shifting Watermarking (GAHSW) Technique Hu et al. (2019) [87]	• GAHSW combined genetic algorithm with a proposed Histogram Shifting of prediction error Watermarking (HSW) approach to reduce distortion as well as improve robustness for database watermarking • GAHSW allows preprocessing, embedding and watermark extraction/ data recovery phases	• Data Set: Forest Cover Type • Workstation with Intel Core i5 CPU at 2.30GHz and 4GB RAM	• Mean absolute error (MAE): 0.746 • Both the original data and the embedded watermark can be recovered efficiently in GAHSW • If an attacker gets to succeed in altering or deleting the tuples up to 90%, GAHSW is capable of recovering minimally half of the watermark information	• This technique can be extended to non-numeric for the shared databases in the distributed environment
A digital watermarking technique for text document protection Khadam et al. (2019) [88]	• The watermark is embedded into different properties of MS-Word document, which can be stored in and shared through the cloud • The secret message is encrypted using a 256-bits AES algorithm and shifted for watermark generation	• Core i3-3110 M CPU @2.40 GHz, RAM-4.0GDDR, Window 10 operating system • MS-Word 2016 and VB 6.0 are used as the development tools • Randomly generated documents, document belongs to the University of Engineering and Technology, Pakistan	• The similarity factor around 99.99 is achieved • The experimental results proved that the scheme is highly imperceptible • The scheme provided the same results in the cloud environment	• This work can be extended for the copyright protection of printed text documents
Joint Watermarking Encryption-Compression (JWEC) scheme for medical images Haddad et al. (2020) [89]	• The accessibility to security services based on the watermarking technique is allowed in the scheme from both compressed and encrypted image bitstreams without passing through decompression or decryption • The scheme integrated the bit-substitution watermarking technique, JPEG-LS, and the AES block cipher algorithm in its cipher block chaining (CBC) mode by performing a single operation on the image which is entirely compressed and encrypted • Decryption, decompression as well as message extraction phases are carried out independently without the requirement of adaption or modification	• Two sets of medical images of 8-bit depth: (i) 100 ultrasound images of 579×690 pixels (ii) 1200 retina images of 627×643 pixels	• In case of the Lena image, the greater values of embedding capacity and PSNR are attained; a PSNR of 42.7 dB is secured by the scheme for a capacity of 0.2 bpp or identically 52757 bits • JWEC minimizes the computati-onal complexity since the decompression and encryption are not required by the scheme in the compressed and encrypted domain respectively for the verification of the image reliability	• Slower than simply lossless JPEG for compressing and encrypting the image

TABLE 4. *(Continued.)* **A capuslization of watermarking based models.**

A separable robust reversible watermarking scheme in encrypted 2D vector graphics Peng et al. (2020) [90]	• For performing the graphics encryption in the polar coordinate system, the polar angles of the vertices are scrambled through a key by the content owner • A watermark is embedded by marginally adjusting the polar angle of the vertex • The encoded watermark partitions are mapped to distinct vertices by the watermark embedder underlying a Hash-based Message Authentication Code (HMAC) function and an embedding key	• The experiments are conducted on a machine with i5- 2520M CPU of 2.50GHz and RAM of 4GB • Visual Studio2015, Teigha_vc14 Libraries, AutoCAD 2016 • 60 self-constructed 2D vector graphics were employed to perform the experiments, out of which 10 have more than 25, 000 vertices	• The average bit error rate (BER) is 0.0044 • The security of the encryption is proved in the scheme by achieving the correlation between X/Y coordinates of the original and the encrypted graphics near to 0	• The data extraction may become failed in case of the attack to the referred vertex • The robustness and the security can be further improved by applying more efficient watermark mapping and coding approaches over the scheme



(a) Worst (b) Average (c) Optimal

FIGURE 7. **Distribution Strategy (a)** $\mathbb{W}_j^* = \mathbb{W}_k^*$ **(b) minimize** $\sum_{j \neq k} |\mathbb{W}_j^* \cap \mathbb{W}_k^*|$ **(c) minimize** $\sum_j \frac{1}{|\mathbb{W}_j^*|} \sum_{k \neq j} |\mathbb{W}_j^* \cap \mathbb{W}_k^*|$.

description regarding the influential models founded on the probability technique is demonstrated in Table 5.

VII. COMPARATIVE AND COMPREHENSIVE ANALYSIS

Table 6 depicts the in-depth analysis of each significant reviewed technique along with its strength and weakness and reflects a comparison among these techniques by accounting for multiple criteria (*CR*). The following are the observations from this table:

- Out of the five techniques Cryptography (*CG*), Access Control (*AC*), Differential Privacy with machine learning (*DP*), Watermarking (*WM*), and Probability (*PB*), privacy (*P*) is assured by *CG* and *DP* only while the security (*S*) is ensured by all the five techniques. It is implied that only *CG* and *DP* preserve both privacy and security among all the five techniques.
- Data leakage prevention (*L*) is procured by *CG*, *AC*, and *DP* while Data leaker detection (*D*) is affirmed by the *WM* and *PB* techniques. It is signified that no technique among all the five techniques provides both prevention and detection simultaneously.
- Out of the three parameters confidentiality (*C*), integrity (*I*), and accessibility (*A*), all the three are achieved by *CG*, *AC*, and *DP* while *WM* and *PB* are capable of obtaining integrity only.
- For the data protection (*DR*) criteria, *CG*, and *DP* outperform the other three techniques *AC*, *WM*, and *PB* since the maximum security parameters are fulfilled by these two techniques, but these two techniques do not enable the leakage detection which is equivalently essential to the other security parameters.
- Out of the two parameters utility (*U*) and sharing (*X*) of data usability (*DU*), it is investigated that *U* is low in the

TABLE 5. A capsulization of probability based models.

Model	Workflow	Implementation	Outcome	Drawbacks & Future Scope
Guilt Agent Model (GAM) Papadimitriou et al. (2011) [103]	• The model detects an agent responsible for leaking the data • The model handles the explicit and sample data request of the user and assessed the guilty of agents using perturbation technique • The method improves the probability of identifying agents by providing data allocation strategies • Provided s-max, s-overlap, s-random, s-sum algorithms for allocation of sample data among agents	• The protocol is implemented in Python with the simulated data leakage problem • Random generated dataset, Number of data objects – 10 & 50, Number of agents – 10 & (2 – 31), Number of requests – 8 & (6 – 15) for explicit request & sample request respectively	• The probability has been computed for various proposed algorithms • It is concluded that the performance of s-max is better as compared to the other algorithms • The method uninfluenced from the amendments in the exposed data such as watermarks	• The fixed number of agents are considered in the model as well as the requests of these agents have acknowledged beforehand which is impracticable in actuality • The future scope includes the investigation of the model that capture leakage scenarios • The requests of the agents can be handled in an online style via extending the presented allocation strategies
Misuseability weight concept Harel et al. (2012) [104]	• The concept estimated the threat originating from the disclosure of data to users that may illegally explore the data • A score is assigned in the approach to the data that may be disclosed to an illegal party, for evaluating the sensitivity level of the disclosed data. Moreover, the evaluated score is utilized to forecast the insider's capability of malevolently exploiting the valuable data	• A four-part questionnaire is designed to conduct the experiment • The knowledge through the specialists is procured in the first two parts • The worth of the established knowledge prototype is evaluated in the last two parts • The collaboration with Deutsche Telekom to hire the domain experts	• Pairwise comparison approach is best in terms of expert time to acquire the knowledge • Records Ranking model is the most accurate for classification accuracy • Useful in several leakage scenarios	• The model gives the estimation of the influential impairments which may occur due to the misused or leaked data but does not protect the data from leakage • The method may lead to an inaccurate result when the knowledge is subjective and dependent on time
Effective data leakage detection through evaluation of algorithms Kumar et al. (2013) [105]	• The allocation strategies are presented by adopting the inspiration from the concept of no wait paradigm where it is not expected from the agents to wait for the allocation of additional agents • The method focuses on minimizing the overlapping of allocated data objects among the agents	• The presented algorithm has been implemented in the .Net framework using C# • The model deals with the sample data requests • Random generated dataset • Number of data objects – 50 • The model considered two cases for the agent's request – i) all the request are same ii) Request may be different	• The probability has been computed for various proposed algorithms • The model worked in the direction of detecting and managing the malicious entity that has leaked the data to an unauthorized third party • The data is distributed by altering the sequence of agents instead of fixing it	• Count of both the data objects and agents are fixed in the model • The method fails when the third party is successful in obtaining the data through stealing or any other means • The strategies of data allocation can be implemented for explicit data requests in the future

TABLE 5. ***(Continued.)*** **A capsulization of probability based models.**

A distribution model for data leakage prevention Fan et al. (2013) [106]	• Distribution model minimizes the overlapping of data distribution • Realized multiple rounds distribution and accounted for the effect due to the guilt probability of users	• Random generated data set in which users are divided into honest (20%), malicious (20%), and common (60%) types	• For the malignant users, the average guilt probability is noticed approximately 0.8 • The deviation of average guilt probabilities among the honest and malicious agents is maximally 0.7	• Simulation data sets are very limited • More probability parameters need to compute
A misuseability measure named TM-Score Vartanian et al. (2014) [107]	• Defined a TM-Score for the textual content that provided a valuation to the impactful harm which is resulted when the information is misused or unintentionally leaked • TM-score is assigned in accordance with the sensitivity and amount of the data	• The inbox folder of 47 employees where an individual folder has $6-25$ emails is randomly selected from Enron email data set • 132 students from the Department of Information Systems Engineering at the Ben-Gurion University as Enron domain expert	• In comparison with the fingerprinting measure, the cosine similarity measure is more efficient • TM-Score is implemented by incorporating a relevantly lesser involvement of the specialist through assigning the labels to $15\% - 20\%$ documents of the repository	• The system is not able to detect and prevent data leakage • The method has no meaning when the information is exchanged through hard copy or by speaking
DLDR (Data Leakage Detection and Reduction) model Sodagudi et al. (2015) [108]	• The model allocated the data through the request-response method for identifying and reducing the data leakage within MANET • Utilized the concept of cryptography and routing protocol implementation at various phases of data transmission	• Real-time data is grabbed from facebook.com • The data in the files are partitioned into the blocks of 64-bits with a key size of 80-bits	• The system characterizes the data loss and minimizes the performance degradation in MANET • Supports data confidentiality	• Does not consider all the possible threats to protect data during sharing • The work includes the expansions of different allocation strategies for handling the data distribution requests by taking into account the vulnerabilities of data loss
Data leakage detection algorithm on the basis of probabilities and task sequences Guevara et al. (2017) [109]	• Identify the behavior of authorized user from their task history carried out on the files of the information system • 2-length sequences are obtained from the identified behavior by codifying it. Furthermore, an algorithm which is adapted from the probability of attained sequences is imposed • The Markov chains are applied for double-checking the activities classified as feasible anomalies before taking the final decision	• Data Set: A government institution of Ecuador delivered the real dataset • 56% i.e. 7138 sessions for testing while 44% i.e. 5589 sessions for training	• Detection rate up to 96.03% • Takes between 4 to 8 milliseconds for used data sets • Computationally efficient	• Works with historical data, which is a time-consuming process • Instead of Markov chain, any other logic can be applied to decide for blocking the system

TABLE 6. A comprehensive analysis and comparison among discussed techniques.

CR		CG	AC	DP	WM	PB
DR	P	✓	×	✓	×	×
	S	✓	✓	✓	✓	✓
	L	✓	✓	✓	×	×
	D	×	×	×	✓	✓
	C	✓	✓	✓	×	×
	I	✓	✓	✓	✓	✓
	A	✓	✓	✓	–	–
DU	U	↓	↓	→	↑	↑
	X	✓	✓	✓	✓	✓
CE	V	✓	✓	×	×	×
	B	✓	✓	✓	✓	✓
	H	✓	✓	×	×	✓
OT	T	✓	×	✓	✓	×
	O	✓	✓	✓	✓	✓
SG		• A robust method for preserving the data confidentiality • Strong prevention mechanism • Strong cryptography can produce maximum security and privacy • Wieldy used and have many options	• Controlled exposure of the confidential data to the semi-honest provider • Convenient for any organization when data classification and access rights are sufficiently accustomed • Minimizes the risk of data leakage	• Preserve both privacy and utility • Controls the information exposure • Prevent the data from leakage • Less dependent on administrative help	• An effective technique that can identify the absolute client responsible for leaking the data • Very strong mechanism for leaker detection from unmodified data • Preserve high data utility • Useful in leaker identification, proof of ownership, and copyright protection	• The detection of a vicious entity is unaffected with the variation in or the absolute destruction of the shared information • Easy to manage • Flexible and adaptable • Effective method for leaker detection • Applicable to every data type
WN		• Can protect the confidential data, yet may not decline its occurrence • Once the key used for cryptography is cracked, then the sensitive	• Unable to identify the vicious entity against data leakage occurrence • Results in reduced data utility	• Produces a considerable amount of overheads • Vulnerable to unauthorized access • Not a detective technique,	• Involves modification in the embedded data • Unable to detect the leaker when the embedded information is destroyed by the malevolent agent	• Provides the estimation only which becomes difficult when the same data is allocated to multiple agents or overlapping of data among agents increases

TABLE 6. ***(Continued.)*** **A comprehensive analysis and comparison among discussed techniques.**

	data can be compromised • Not capable to identify the vicious party • Does not detect data leak • Produces a considerable amount of overheads	• Affected by improper data classification and access control policy • Dependent on administrative help	consequently, the technique is ineffectual in case a leakage has happened • Applicable to statistical information only • Limited to specific situations and scenarios	• Results into wrong identification when the watermark is altered • Certain data cannot admit watermarks • Cannot prevent the data leakage	• Does not provide a preventive mechanism • Unable to obstruct unauthorized access for acquiring and maltreating the data

CR- Criteria; **CG**- Cryptography; **AC**- Access Control; **DP**- Differential Privacy with machine learning; **WM**- Watermarking; **PB**- Probability; **DR**- Data Protection; **P**- Privacy; **S**- Security, **L**- Data Leakage Prevention; **D**- Data Leaker Detection; **C**- Confidentiality, **I**- Integrity; **A**- Accessibility; **DU**- Data Usability; **U**- Utility; **X**- Sharing; **CE**- Cloud Environment; **V**- Private; **B**- Public; **H**- Hybrid; **OT**- Other; **T**- Transformation; **O**- Overhead; **SG**- Strength; **WN**- Weakness; ↑- High; →- Moderate; ↓- Low; ✓- Yes, ×- No; – Not exist

case of *CG* and *AC*, moderate in the case of *DP*, and high for *WM* and *PB* techniques while the *X* is assured by all the five techniques. Hence, the performance of *WM* and *PB* is high for the criteria *DU* compares to the other three techniques.

- Out of the three parameters private (*V*), public (*B*), and hybrid (*H*) of the cloud environment (*CE*), the models based on *CG* and *AC* techniques exist for private cloud, the model based on all the five techniques persist for public cloud and the model based on *CG*, *AC* and *PB* techniques exist for hybrid cloud. It is indicated that out of the five techniques, *CG* and *AC* techniques are utilized for all three types of the cloud.
- In the case of other (*OT*) criteria, *CG*, *DP*, and *WM* comprise data transformation. Data is transformed into ciphertext, noised data, and watermarked data in the *CG*, *DP*, and *WM* techniques respectively while *AC* and *PB* techniques do not engage any data transformation which results in less computation complexity compared to *CG*, *DP*, and *WM* techniques. Furthermore, all the five techniques involve overhead for protecting the data due to the transformation of data before communicating in the case of *CG*, *DP*, and *WM* techniques, requisition for the formulation of access control policies (ACPs) in the case of *AC* technique while *PB* implies the overhead in the form of probability computation for malicious entity identification. It is reported that *AC* and *PB* techniques outperform the other three techniques for the parameter *OT*.
- Although *CG* is a powerful technique for preserving the data privacy, security, confidentiality, and leakage prevention, this technique has a crucial drawback that once the key applied is revealed, then the data can be compromised. Also, the technique is unable to identify the culprit entity and consists of a considerable amount of overheads.
- The benefit of the *AC* technique is that it enables monitored disclosure of the sensitive data and minimizes the risk of data leakage without the involvement of transformation cost but the technique is incapable of identifying the vicious party in case of the data leakage occurrence and does not ensure data privacy.
- *DP* technique has the advantage that it controls the information exposure, preserves privacy, security as well as the utility, and prevents the data from leakage, but this technique is ineffective for malicious entity identification. It involves a substantial amount of overhead. Also, this technique is not fit for applications where absolute data is required without any modifications.
- *WM* is a sturdy leaker detection technique that preserves high utility along with security to the data. The strength of the technique is its effectiveness in detecting the absolute entity that revealed the confidential information. However, the technique has a crucial limitation of becoming incapable of identifying the malignant entity when the embedded information is completely removed or altered by the vicious agent. This technique cannot prevent data leakage, comprises of overhead, and does not assure privacy. Also, the technique may not be applicable to every data type.
- The effectiveness of the *PB* technique is that the identification of the malignant entity is not impacted by the amendments in the shared data, unlike the watermarking technique. Also, data security is not key-dependent in the *PB* technique, unlike the cryptography technique where the key can be compromised. It is a powerful leaker detection technique that is applicable to every data type and ensures high utility in addition to the security

without involving the transformation cost. But this technique constitutes an estimation only of the malicious entity and does not assure privacy. Also, the technique does not support the leakage preventive mechanism and cannot cease illegal access to attain and mistreat the data.

Comprehensively, it is inferred that *CG* is leading among the five techniques for preserving privacy, security, confidentiality as well as leakage prevention. *AC* is the foremost technique to ensure privacy without involving the cost of transformation. *DP* is the prime technique to preserve privacy coupled with utility. Watermarking and probability are the best techniques for assuring data utility in association with leaker detection. Furthermore, *WM* is the optimum leaker detection technique for absolute culprit identification while *PB* is the superior leaker estimation technique without having the impact of data modification. However, no technique alone is sufficient to provide completely secure methodologies and there arises a necessity to utilize an integration of the techniques for an effective data protection mechanism.

VIII. CONCLUSION AND FUTURE WORK

Data protection is a challenging task in the field of cloud computing and information security. A plethora of work is interpreted to mitigate this challenge. However, there is an inadequacy for the comprehensive study of the ongoing solutions. From this perspective, this paper presented a comprehensive analysis and explored the foremost techniques concerning the functionality and the relevant solutions to share the data securely for data protection in the cloud environment. The essential and adequate information which is desired to fetch the core of the method along with the research gaps and future directions about each discussed solution is highlighted. Furthermore, exhaustive analysis and a comparison among the refereed techniques are performed. The relevancy of every technique is analyzed in compliance with the context.

It is investigated that no technique alone is efficient in ensuring the absolute security of the data from every directly or indirectly engaged party in the system. The robust solution can be developed by integrating the techniques for providing complete security to the system in the sharing environment. Moreover, with the set of highlights of addressed remarkable solutions, it is deemed that the exposed analysis will act as a milestone for the potential researchers working in the area as well as other emerging applications demanding secure data storage and sharing for its protection.

REFERENCES

[1] A. K. Singh and I. Gupta, "Online information leaker identification scheme for secure data sharing," *Multimedia Tools Appl.*, vol. 79, no. 41, pp. 31165–31182, Nov. 2020.
[2] E. Zaghloul, K. Zhou, and J. Ren, "P-MOD: Secure privilege-based multilevel organizational data-sharing in cloud computing," *IEEE Trans. Big Data*, vol. 6, no. 4, pp. 804–815, Dec. 2020.
[3] I. Gupta and A. K. Singh, "GUIM-SMD: Guilty user identification model using summation matrix-based distribution," *IET Inf. Secur.*, vol. 14, no. 6, pp. 773–782, Nov. 2020.
[4] W. Shen, J. Qin, J. Yu, R. Hao, and J. Hu, "Enabling identity-based integrity auditing and data sharing with sensitive information hiding for secure cloud storage," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 2, pp. 331–346, Feb. 2019.
[5] I. Gupta and A. K. Singh, "An integrated approach for data leaker detection in cloud environment," *J. Inf. Sci. Eng.*, vol. 36, no. 5, pp. 993–1005, Sep. 2020.
[6] R. Li, C. Shen, H. He, X. Gu, Z. Xu, and C.-Z. Xu, "A lightweight secure data sharing scheme for mobile cloud computing," *IEEE Trans. Cloud Comput.*, vol. 6, no. 2, pp. 344–357, Apr. 2018.
[7] I. Gupta, N. Singh, and A. K. Singh, "Layer-based privacy and security architecture for cloud data sharing," *J. Commun. Softw. Syst.*, vol. 15, no. 2, pp. 173–185, Apr. 2019.
[8] J. Li, S. Wang, Y. Li, H. Wang, H. Wang, H. Wang, J. Chen, and Z. You, "An efficient attribute-based encryption scheme with policy update and file update in cloud computing," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6500–6509, Dec. 2019.
[9] C. Suisse. (2017). *2018 Data Center Market Drivers: Enablers Boosting Enterprise Cloud Growth*. Accessed: May 19, 2019. [Online]. Available: https://cloudscene.com/news/2017/12/2018-data-center-predictions/
[10] I. Gupta and A. K. Singh, "A framework for malicious agent detection in cloud computing environment," *Int. J. Adv. Sci. Technol.*, vol. 135, pp. 49–62, Feb. 2020.
[11] Y. Li, Y. Yu, G. Min, W. Susilo, J. Ni, and K.-R. Choo, "Fuzzy identity-based data integrity auditing for reliable cloud storage systems," *IEEE Trans. Dependable Secure Comput.*, vol. 16, no. 1, pp. 72–83, Jan./Feb. 2019.
[12] I. Gupta and A. K. Singh, "A probabilistic approach for guilty agent detection using bigraph after distribution of sample data," *Proc. Comput. Sci.*, vol. 125, pp. 662–668, Jan. 2018.
[13] L. Zhang, Y. Cui, and Y. Mu, "Improving security and privacy attribute based data sharing in cloud computing," *IEEE Syst. J.*, vol. 14, no. 1, pp. 387–397, Mar. 2020.
[14] I. Gupta and A. K. Singh, "Dynamic threshold based information leaker identification scheme," *Inf. Process. Lett.*, vol. 147, pp. 69–73, Jul. 2019.
[15] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1265–1277, Jun. 2016.
[16] I. Gupta and A. K. Singh, "SELI: Statistical evaluation based leaker identification stochastic scheme for secure data sharing," *IET Commun.*, vol. 14, no. 20, pp. 3607–3618, Dec. 2020.
[17] W. Teng, G. Yang, Y. Xiang, T. Zhang, and D. Wang, "Attribute-based access control with constant-size ciphertext in cloud computing," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 617–627, Oct./Dec. 2017.
[18] I. Gupta and A. K. Singh, "A probability based model for data leakage detection using bigraph," in *Proc. 7th Int. Conf. Commun. Netw. Secur. (ICCNS)*. New York, NY, USA: Assoc. Comput. Machinery, 2017, pp. 1–5.
[19] L. Columbus. (Jan. 2018). *83% of Enterprise Workloads Will Be in the Cloud by 2020*. [Online]. Available: https://www.forbes.com/sites/louiscolumbus/2018/01/07/83-of-enterprise-workloads-will-be-in-the-cloud-by-2020/#50d375286261
[20] Gartner. (2018). *Gartner Forecasts Worldwide Public Cloud Revenue to Grow 17.3 Percent in 2019*. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2018-09-12-gartner-forecasts-worldwide-public-cloud-revenue-to-grow-17-percent-in-2019
[21] (2019). *Cloud IT Infrastructure Revenues Surpassed Traditional IT Infrastructure Revenues for the First Time in the Third Quarter of 2018*. Accessed: May 19, 2019. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS44670519
[22] (Dec. 2021). *Alarming Cyber Security Facts to Know for 2021 and Beyond*. [Online]. Available: https://www.cybertalk.org/2021/12/02/alarming-cyber-security-facts-to-know-for-2021-and-beyond/
[23] W. Li, K. Xue, Y. Xue, and J. Hong, "TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 5, pp. 1484–1496, May 2016.
[24] X. Ma, J. Ma, H. Li, Q. Jiang, and S. Gao, "PDLM: Privacy-preserving deep learning model on cloud with multiple keys," *IEEE Trans. Services Comput.*, vol. 14, no. 4, pp. 1251–1263, Jul. 2021.
[25] I. Gupta and A. K. Singh, "A confidentiality preserving data leaker detection model for secure sharing of cloud data using integrated techniques," in *Proc. 7th Int. Conf. Smart Comput. Commun. (ICSCC)*. Sarawak, Malaysia: Curtin Univ., Jun. 2019, pp. 1–5.

[26] S. Xu, G. Yang, Y. Mu, and R. H. Deng, "Secure fine-grained access control and data sharing for dynamic groups in the cloud," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 8, pp. 2101–2113, Aug. 2018.

[27] Z. Zhu and R. Jiang, "A secure anti-collusion data sharing scheme for dynamic groups in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 40–50, Jan. 2016.

[28] Cyber Risk Analytics (CRA) and Risk Based Security (RBS). (2021). *2021 Data Breach Quick View Report*. [Online]. Available: https://pages.riskbasedsecurity.com/hubfs/Reports/2021/2021%20Year%20End%20Data%20Breach%20QuickView%20Report.pdf

[29] A Study Conducted by Ponemon Institute and Sponsored, Analyzed, Reported by IBM Security. (Jul. 2021). *2021 Cost of a Data Breach Report*. [Online]. Available: https://branden.biz/wp-content/uploads/2021/08/Cost-of-af-DATA-Breach-Report-2021.pdf

[30] Y. Kao, K. Huang, H. Gu, and S. Yuan, "UCloud: A user-centric key management scheme for cloud data protection," *IET Inf. Secur.*, vol. 7, no. 2, pp. 144–154, Jun. 2013.

[31] A. Al-Haj, G. Abandah, and N. Hussein, "Crypto-based algorithms for secured medical image transmission," *IET Inf. Secur.*, vol. 9, no. 6, pp. 365–373, Nov. 2015.

[32] K. Liang, M. H. Au, J. K. Liu, W. Susilo, D. S. Wong, G. Yang, Y. Yu, and A. Yang, "A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing," *Future Generat. Comput. Syst.*, vol. 52, pp. 95–108, Nov. 2015.

[33] H. Liu, X. Li, M. Xu, R. Mo, and J. Ma, "A fair data access control towards rational users in cloud storage," *Inf. Sci.*, vols. 418–419, pp. 258–271, Dec. 2017.

[34] Z. Liu, Z. L. Jiang, X. Wang, and S. M. Yiu, "Practical attribute-based encryption: Outsourcing decryption, attribute revocation and policy updating," *J. Netw. Comput. Appl.*, vol. 108, pp. 112–123, Apr. 2018.

[35] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2007, pp. 321–334.

[36] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in *Proc. 17th ACM Conf. Comput. Commun. Secur.*, New York, NY, USA, 2010, pp. 735–737.

[37] G. Wang, Q. Liu, J. Wu, and M. Guo, "Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers," *Comput. Secur.*, vol. 30, no. 5, pp. 320–331, 2011.

[38] K. Yang, X. Jia, and K. Ren, "Secure and verifiable policy update outsourcing for big data access control in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 12, pp. 3461–3470, Dec. 2015.

[39] S. Kamara and K. Lauter, "Cryptographic cloud storage," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, vol. 6054, 2010, pp. 136–149.

[40] J. Dai and Q. Zhou, "A PKI-based mechanism for secure and efficient access to outsourced data," in *Proc. Int. Conf. Netw. Digit. Soc.*, vol. 1, May 2010, pp. 640–643.

[41] S. Sanka, C. Hota, and M. Rajarajan, "Secure data access in cloud computing," in *Proc. IEEE 4th Int. Conf. Internet Multimedia Services Archit. Appl.*, Dec. 2010, pp. 1–6.

[42] C. Curino, E. Jones, R. Popa, N. Malviya, E. Wu, S. Madden, H. Balakrishnan, and N. Zeldovich, "Relational cloud: A database-as-a-service for the cloud," in *Proc. 5th Biennial Conf. Innov. Data Syst. Res. (CIDR)*, Asilomar, CA, USA, Jan. 2011, pp. 235–240.

[43] T. C. Bressoud and F. B. Schneider, "Hypervisor-based fault tolerance," in *Proc. 15th ACM Symp. Operating Syst. Princ. (SOSP)*. New York, NY, USA: Assoc. Comput. Machinery, 1995, pp. 1–11.

[44] S. Bajikar, "Trusted platform module (TPM) based security on notebook PCs," Mobile Platforms Group Intel Corp., Santa Clara, CA, USA, White Paper, Jun. 2002, pp. 1–20.

[45] J. M. McCune, Y. Li, N. Qu, Z. Zhou, A. Datta, V. Gligor, and A. Perrig, "TrustVisor: Efficient TCB reduction and attestation," in *Proc. IEEE Symp. Secur. Privacy*, May 2010, pp. 143–158.

[46] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *Advances in Cryptology—EUROCRYPT*, K. Nyberg, Ed. Berlin, Germany: Springer, 1998, pp. 127–144.

[47] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. 13th ACM Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, 2006, pp. 89–98.

[48] L. O. M. Kobayashi, S. S. Furuie, and P. S. L. M. Barreto, "Providing integrity and authenticity in DICOM images: A novel approach," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 4, pp. 582–589, Jul. 2009.

[49] A. Al-Haj, "Providing integrity, authenticity, and confidentiality for header and pixel data of DICOM images," *J. Digit. Imag.*, vol. 28, no. 2, pp. 179–187, Apr. 2015.

[50] A. Lewko and B. Waters, "New proof methods for attribute-based encryption: Achieving full security through selective techniques," in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 2012, pp. 180–198.

[51] P. K. Tysowski and M. A. Hasan, "Hybrid attribute- and re-encryption-based key management for secure and scalable mobile applications in clouds," *IEEE Trans. Cloud Computing*, vol. 1, no. 2, pp. 172–186, Jul. 2013.

[52] A. De Caro and V. Iovino, "JPBC: Java pairing based cryptography," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2011, pp. 850–855.

[53] J. Wang. (2015). *Ciphertext-Policy Attribute Based Encryption Java Toolkit*. [Online]. Available: https://github.com/junwei-wang/cpabe

[54] M. Nabeel and E. Bertino, "Privacy preserving delegated access control in public clouds," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2268–2280, Sep. 2014.

[55] M. Ali, S. U. R. Malik, and S. U. Khan, "DaSCE: Data security for cloud environment with semi-trusted third party," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 642–655, Oct. 2017.

[56] A. Almutairi, M. I. Sarfraz, and A. Ghafoor, "Risk-aware management of virtual resources in access controlled service-oriented cloud datacenters," *IEEE Trans. Cloud Comput.*, vol. 6, no. 1, pp. 168–181, Jan. 2018.

[57] J. Hong, K. Xue, Y. Xue, W. Chen, D. S. L. Wei, N. Yu, and P. Hong, "TAFC: Time and attribute factors combined access control for time-sensitive data in public cloud," *IEEE Trans. Services Comput.*, vol. 13, no. 1, pp. 158–171, Jan. 2020.

[58] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography—PKC*. Berlin, Germany: Springer, 2011, pp. 53–70.

[59] Z. Wan, J. Liu, and R. H. Deng, "HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 743–754, Apr. 2012.

[60] H. Deng, Q. Wu, B. Qin, J. Domingo-Ferrer, L. Zhang, J. Liu, and W. Shi, "Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts," *Inf. Sci.*, vol. 275, pp. 370–384, Aug. 2014.

[61] Y. Tang, P. P. C. Lee, John C. S. Lui, and R. Perlman, "Secure overlay cloud storage with access control and assured deletion," *IEEE Trans. Dependable Secure Comput.*, vol. 9, no. 6, pp. 903–916, Nov./Dec. 2012.

[62] A. Sahai, H. Seyalioglu, and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," in *Advances in Cryptology—CRYPTO*, R. Safavi-Naini and R. Canetti, Eds. Berlin, Germany: Springer, 2012, pp. 199–217.

[63] K. Lee, "Ciphertext outdate attacks on the revocable attribute-based encryption scheme with time encodings," *IEEE Access*, vol. 7, pp. 165122–165126, 2019.

[64] E. Androulaki, C. Soriente, L. Malisa, and S. Capkun, "Enforcing location and time-based access control on cloud-stored data," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst.*, Jun. 2014, pp. 637–648.

[65] R. Yonetani, V. N. Boddeti, K. M. Kitani, and Y. Sato, "Privacy-preserving visual learning using doubly permuted homomorphic encryption," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2040–2050.

[66] E. Hesamifard, H. Takabi, M. Ghasemi, and N. W. Rebecca, "Privacy-preserving machine learning as a service," *Proc. Privacy Enhancing Technol.*, vol. 2018, no. 3, pp. 123–142, Jun. 2018.

[67] P. Li, J. Li, Z. Huang, C.-Z. Gao, W.-B. Chen, and K. Chen, "Privacy-preserving outsourced classification in cloud computing," *Cluster Comput.*, vol. 21, no. 1, pp. 277–286, Mar. 2018.

[68] T. Li, Z. Huang, P. Li, Z. Liu, and C. Jia, "Outsourced privacy-preserving classification service over encrypted data," *J. Netw. Comput. Appl.*, vol. 106, pp. 100–110, Mar. 2018.

[69] P. Li, T. Li, H. Ye, J. Li, X. Chen, and Y. Xiang, "Privacy-preserving machine learning with multiple data providers," *Future Gener. Comput. Syst.*, vol. 87, pp. 341–350, Oct. 2018.

[70] C.-Z. Gao, Q. Cheng, P. He, W. Susilo, and J. Li, "Privacy-preserving naive Bayes classifiers secure against the substitution-then-comparison attack," *Inf. Sci.*, vol. 444, pp. 72–88, May 2018.

[71] T. Li, J. Li, Z. Liu, P. Li, and C. Jia, "Differentially private Naive Bayes learning over multiple data sources," *Inf. Sci.*, vol. 444, pp. 89–104, May 2018.

[72] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A Matlab-like environment for machine learning," in *Proc. NIPS Workshop*, Jan. 2011, pp. 1–6.

[73] Y. Li, V. Swarup, and S. Jajodia, "Fingerprinting relational databases: Schemes and specialties," *IEEE Trans. Dependable Secure Comput.*, vol. 2, no. 1, pp. 34–45, Jan. 2005.
[74] J. Kiernan, R. Agrawal, and P. J. Haas, "Watermarking relational data: Framework, algorithms and analysis," *VLDB J. Int. J. Very Large Data Bases*, vol. 12, no. 2, pp. 157–169, Aug. 2003.
[75] M. Shehab, E. Bertino, and A. Ghafoor, "Watermarking relational databases using optimization-based techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 1, pp. 116–129, Jan. 2008.
[76] S. Bishop, H. Okhravi, S. Rahimi, and Y. C. Lee, "Covert channel resistant information leakage protection using a multi-agent architecture," *IET Inf. Secur.*, vol. 4, no. 4, pp. 233–247, Dec. 2010.
[77] N. Kumar, V. Katta, H. Mishra, and H. Garg, "Detection of data leakage in cloud computing environment," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Nov. 2014, pp. 803–807.
[78] S. E. Jero and P. Ramu, "Curvelets-based ECG steganography for data security," *Electron. Lett.*, vol. 52, no. 4, pp. 283–285, 2016.
[79] M. Backes, N. Grimm, and A. Kate, "Data lineage in malicious environments," *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 2, pp. 178–191, Mar. 2016.
[80] D. Ulybyshev, B. Bhargava, and A. Oqab-Alsalem, "Secure data exchange and data leakage detection in an untrusted cloud," in *Applications of Computing and Communication Technologies*. Singapore: Springer, 2018, pp. 99–113.
[81] A. Marzieh, H. Sadreazami, M. O. Ahmad, and M. N. S. Swamy, "A channel-dependent statistical watermark detector for color images," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 65–73, Jan. 2019.
[82] J. R. Hernandez, M. Amado, and F. Perez-Gonzalez, "DCT-domain watermarking techniques for still images: Detector performance analysis and a new structure," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 55–68, Jan. 2000.
[83] R. Kwitt, P. Meerwald, and A. Uhl, "Color-image watermarking using multivariate power-exponential distribution," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 4245–4248.
[84] H. Sadreazami, M. O. Ahmad, and M. N. S. Swamy, "A robust multiplicative watermark detector for color images in sparse domain," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 12, pp. 1159–1163, Dec. 2015.
[85] M. Rabizadeh, M. Amirmazlaghani, and M. Ahmadian-Attari, "A new detector for contourlet domain multiplicative image watermarking using Bessel K form distribution," *J. Vis. Commun. Image Represent.*, vol. 40, pp. 324–334, Oct. 2016.
[86] M. Amini, H. Sadreazami, M. O. Ahmad, and M. N. S. Swamy, "Multichannel color image watermark detection utilizing vector-based hidden Markov model," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 1–4.
[87] D. Hu, D. Zhao, and S. Zheng, "A new robust approach for reversible database watermarking with distortion control," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 6, pp. 1024–1037, Jun. 2018.
[88] U. Khadam, M. M. Iqbal, M. A. Azam, S. Khalid, S. Rho, and N. Chilamkurti, "Digital watermarking technique for text document protection using data mining analysis," *IEEE Access*, vol. 7, pp. 64955–64965, 2019.
[89] S. Haddad, G. Coatrieux, A. Moreau-Gaudry, and M. Cozic, "Joint watermarking-encryption-JPEG-LS for medical image reliability control in encrypted and compressed domains," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 2556–2569, 2020.
[90] F. Peng, W.-Y. Jiang, Y. Qi, Z.-X. Lin, and M. Long, "Separable robust reversible watermarking in encrypted 2D vector graphics," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2391–2405, Aug. 2020.
[91] R. Sion, M. Atallah, and S. Prabhakar, "Rights protection for relational data," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 12, pp. 1509–1525, Dec. 2004.
[92] (2014). *Pairing-Based Cryptography library (PBC)*. [Online]. Available: http://crypto.stanford.edu/pbc
[93] (2014). *GNU Multiple Precision Arithmetic Library (GMP)*. [Online]. Available: http://gmplib.org/
[94] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," in *Advances in Cryptology—ASIACRYPT*, vol. 2248. Berlin, Germany: Springer, 2001, pp. 514–532.
[95] W. Dai. (2013). *Crypto++ Library*. [Online]. Available: http://cryptopp.com
[96] I. J. Cox, J. Kilian, T. Shamoon, and F. T. Leighton, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1995.
[97] P. Meerwald. (2010). *Watermarking Toolbox*. [Online]. Available: http://www.cosy.sbg.ac.at/pmeerw/Watermarking/source
[98] D. Ulybyshev, B. Bhargava, L. Li, J. Kobes, D. Steiner, H. Halpin, B. C. An, M. Villarreal, R. Ranchal, and T. Vincent, "Secure dissemination of EHR in untrusted cloud," Purdue Univ., West Lafayette, IN, USA, Project Tutorial, 2016.
[99] H. Shacham and B. Waters, "Compact proofs of retrievability," *J. Cryptol.*, vol. 26, no. 3, pp. 442–483, Jul. 2013.
[100] H. Wang, "Proxy provable data possession in public clouds," *IEEE Trans. Services Comput.*, vol. 6, no. 4, pp. 551–559, Oct./Dec. 2013.
[101] B. Wang, B. Li, and H. Li, "Panda: Public auditing for shared data with efficient user revocation in the cloud," *IEEE Trans. Serv. Comput.*, vol. 8, no. 1, pp. 92–106, Jan./Feb. 2015.
[102] J. Shen, J. Shen, X. Chen, X. Huang, and W. Susilo, "An efficient public auditing protocol with novel dynamic structure for cloud data," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 10, pp. 2402–2415, Oct. 2017.
[103] P. Papadimitriou and H. Garcia-Molina, "Data leakage detection," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 1, pp. 51–63, Jan. 2011.
[104] A. Harel, A. Shabtai, L. Rokach, and Y. Elovici, "*M*-score: A misuseability weight measure," *IEEE Trans. Dependable Secure Comput.*, vol. 9, no. 3, pp. 414–428, May 2012.
[105] A. Kumar, A. Goyal, A. Kumar, N. K. Chaudhary, and S. S. Kamath, "Comparative evaluation of algorithms for effective data leakage detection," in *Proc. IEEE Conf. Inf. Commun. Technol.*, Apr. 2013, pp. 177–182.
[106] Y. Fan, Y. Rongwei, W. Lina, and M. Xiaoyan, "A distribution model for data leakage prevention," in *Proc. Int. Conf. Mech. Sci., Electr. Eng. Comput. (MEC)*, Dec. 2013, pp. 2617–2620.
[107] A. Vartanian and A. Shabtai, "*TM*-score: A misuseability weight measure for textual content," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2205–2219, Dec. 2014.
[108] S. Sodagudi and R. R. Kurra, "An approach to identify data leakage in secure communication," in *Proc. 2nd Int. Conf. Intell. Comput. Appl.* Singapore: Springer, 2017, pp. 31–43.
[109] C. Guevara, M. Santos, and V. López, "Data leakage detection algorithm based on task sequences and probabilities," *Knowl.-Based Syst.*, vol. 120, pp. 236–246, Mar. 2017.
[110] P. D. Ezhilchelvan and I. Mitrani, "Evaluating the probability of malicious co-residency in public clouds," *IEEE Trans. Cloud Comput.*, vol. 5, no. 3, pp. 420–427, Jul. 2017.

ISHU GUPTA (Member, IEEE) received the B.C.A. and M.C.A. (Hons.) degrees in computer science from Kurukshetra University, Kurukshetra, India, in 2012 and 2015, respectively, and the Ph.D. degree from the Department of Computer Applications, National Institute of Technology (NIT), Kurukshetra, in 2021. She is currently a Postdoctoral Research Fellow with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan. She has more than 40 publications in international journals and conferences of high repute. Her major research interests include cloud computing, the Internet of Things (IoT), data science, big data, machine learning, information security and privacy, and quantum computing. She is a member of multiple prestigious IEEE and ACM-SIGCOMM Societies. She is awarded the Senior Research Fellowship (SRF) from the University Grants Commission (UGC), Ministry of Human Resource Development (MHRD), Government of India. She received the Excellent Paper Award twice. She received the Gold Medal for her M.C.A. degree.

ASHUTOSH KUMAR SINGH (Senior Member, IEEE) received the Ph.D. degree in electronics engineering from the Indian Institute of Technology (IIT), BHU, Varanasi, India, in 2000. He did his postdoctoral research at the Department of Computer Science, University of Bristol, U.K. He is currently working as a Professor and the Head with the Department of Computer Applications, National Institute of Technology, Kurukshetra, India. He has more than 20 years of research and teaching experience in various universities in India, U.K., and Malaysia. He has authored/coauthored more than 320 research articles and 12 books. His research interests include data science, cloud computing, the Internet of Things (IoT), machine learning, big data, information security and privacy, and quantum computing.

CHUNG-NAN LEE (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from the National Cheng Kung University, Tainan, Taiwan, in 1980 and 1982, respectively, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, USA, in 1992. Since 1992, he has been with the National Sun Yat-sen University, Kaohsiung, Taiwan, where he was the Chairperson of the Department of Computer Science and Engineering, from August 1999 to July 2001, and currently he is a Distinguished Professor and the Director of the Cloud Computing Research Center. His current research interests include multimedia over wireless networks, cloud computing, and the Internet of Things (IoT). He was the President of the Taiwan Association of Cloud Computing, from 2015 to 2017. He was the Vice President for TA of APSIPA, from 2019 to 2020. In 2016, he received an Outstanding Engineering Professor from the Chinese Institute of Engineers, Taiwan.

RAJKUMAR BUYYA (Fellow, IEEE) received the Ph.D. degree in computer science and software engineering from Monash University, Melbourne, VIC, Australia, in 2002. He is currently a Redmond Barry Distinguished Professor and the Director of the Cloud Computing and Distributed Systems (CLOUDS) Laboratory, The University of Melbourne, Melbourne. He is also serving as the Founding CEO at Manjrasoft, a spin-off company of the university, commercializing its innovations in cloud computing. He has authored/coauthored more than 1050 publications and seven textbooks including *Mastering Cloud Computing* published by McGraw Hill, China Machine Press, and Morgan Kaufmann for Indian, Chinese, and international markets, respectively. He is one of the highly cited authors in computer science and software engineering worldwide (H-index=155, G-index=336, and more than 125,954 citations).