International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

Dynamic Searchable Encryption Scheme :Review

Ibtihal Ali khanjar, Haider K. Hoomod, Intisar Abd Yousif

Education College, AlMustansiriyah University, Baghdad, Iraq
Education College, AlMustansiriyah University , Baghdad,Iraq
Education College, AlMustansiriyah University, Baghdad,Iraq

Date of Submission: 01-09-2022 Date of Acceptance: 08-09-2022

ABSTRACT—As cloud computing expands, more and more clients are outsourcing data to the cloud, where it is kept on cloud servers. Privacy is provided by encrypting the data before outsourcing, and the encrypted data is then stored in the cloud. Since the data in this instance is encrypted, searching the ciphertext for the data is challenging. To address this issue, searchable encryption techniques are created. Clients can efficiently explore their outsourced data without compromising the privacy of the documents or search queries while still outsourcing their encrypted documents to a remote server. Data addition, deletion, and file alteration are all possible with dynamic searchable symmetric encryption following data outsourcing. The forward and backward privacy are two significant privacy that are addressed by DSSE methods. The first one makes sure to keep the recently uploaded documents from being associated with earlier search queries. The second, made certain that the deleted documents cannot be connected to later search requests. In this study, we reviewed the various DSSE techniques currently in the cloud industry. We will demonstrate how this scheme has evolved as well as the problems and techniques that have been applied to various DSSE concerns in order to achieve secrecy and efficiency in many areas, giving the reader and the beginner a reference point for their next work.

Keywords—Forward. Backward. Cloud Computing. Leakage

I. INTRODUCTION

Big data has created a need for cloud storage, and with cloud computing experiencing tremendous growth, more and more users are choosing to outsource to store their data to servers. Cloud administrators can easily access user data if it is kept without any encryption. The user's unencrypted data could also be accessed by hackers in the event of an attack. The most typical solution to this issue is to encrypt data using general symmetric encryption prior to outsourcing in order to protect the secrecy of the data that was outsourced. The natural structure of data is destroyed by encryption, though, and keyword-based search services become unusable [1]. Searchable symmetric encryption (SSE) [2] [3] proposed executing a safe search through data encryption as a solution to this conundrum. Early SSE schemas were only suitable for static adjustment, which constrained their use. To make the plan more workable, Dynamic Symmetric Search Scheme (DSSE) A variety of various DSSE schemes have been created in the past that not only satisfy the fundamental requirements of adding and removing data but also have distinctive features. The DSSE technique is immune to adaptive chosen-keyword attacks and can finish the search in a sublinear time. They was first presented by S. Kamara, et al.[4] and Cash et al.[5] proposed the first DSSE that was utilized on huge databases in 2014. Their plan offers data dynamics that let the user update data over time without compromising data privacy or searchability, and it offers a quick and secure way to add and remove data from an encrypted index structure. Owing to the potential for more information to be disclosed to the server due to updates, it might be difficult to create secure DSE schemes. Forward and backward privacy are two security concepts relevant to the security of DSSE schemes. A technique offers forward privacy when the addiction process doesn't reveal any information about the additional document and its keywords. When two keyword searches are separated by a backward privacy DSSE, no information regarding the documents that were added and removed is revealed.

The majority of traditional DSSE schemes are vulnerable to various assaults and leak searches. There were several attacks displayed. Several DSSE plans have been put up to address these leaks, the majority of which are in the

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

forward privacy, and Creating secure DSSE schemas is another way to prevent information from leaking, and doing so exposes users to the risk of file injection attacks during the insertion and deletion operations. [4] But the majority of these DSSE designs lacked effectiveness and security. ORAM technologies allow for the concealment of search access patterns.[6] due to its extremely high costs, this technology has hardly advanced at all. using incredibly effective techniques to conceal these leaks as well as the access pattern and search pattern.

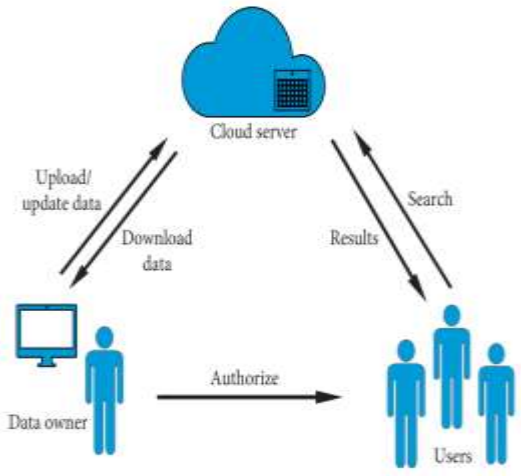
.

II. OUR CONTRIBUTION

The main contribution of this study is a review of dynamic searchable encryption in cloud computing, which offers the following:

1. DSSE schemes are evaluated in terms of their search capabilities, including multi-keyword and attribute-based keyword searches, multi-keyword ranked encrypted searches and single keyword searches in a select few DSSE
2. Based on index generation time, search time, and update time, the performance of various DSSE methods is examined.
3. The security model of DSSE schemes and the various attacks that could be made against them are described.
4. Examined the DSSE scheme's difficulties in achieving forward and backward privacy.

For the majority of our contributions, we will offer earlier research and the recommendations made in the section review.



.

Figure 1: System model in DSSE

III. SYSTEM MODEL

We show the model of the system in Figure 1, which consists of three objects:

1. Data Owner (DO). First he created secure encrypted index tree that is safe and easy to search from document and the collection of documents that he wants to send to the cloud server. His encrypted EDB is then sent to the cloud server after being encrypted for all files and the index using various private keys. He is in charge of updating files and managing users in addition to these responsibilities. the data owner is always considered to be reliable.

2.Cloud Server (CS). The cloud server's primary function is to store encrypted files and index from DO while also performing searches for DU. It performs searches over the index and provides users with the search results. The cloud server is viewed as a trustworthy yet curious entity.

3.Data User (DU). The data owners upload the authorized users of the document collection. the data user creates searches for the file using a multiple of keywords. the search token is sent to the (CS), where they then obtain the search results. The users utilize the shared secret keys to decrypt the encrypted data after receiving it.

IV. METRICS TO ACHIEVE AND CREATE SECURE DSSE SCHEME

The metrics are still an open research challenge that must be resolved to make a fully secure DSSE scheme in Real-world applications. And we will present what was presented and suggested by researchers to achieve some of these metrics [7][8]

- Dynamic. The scheme allows the client to update the encryption collection documents after their uploading to the cloud server.
- Storage size at client and cloud server.
- Parallelization of computation.
- Efficient search and update time and computation complexity
- Secure search. The scheme prevents the server from learning or obtaining any information about the document set and queries.

V. LITERATURE REVIEW

This section reviews some important advancements made in the previous years before giving a brief overview of the current schemes in (DSSE). This paper's constrained length prevents it from covering the complete corpus of research in the area. The review's focus is constrained and closely relates to the authors' personal areas of interest.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

In 2015, W. Luo et al.[9] they presented Dynamic Searchable Symmetric Encryption scheme with multi-user has been proposed that the organization authorizes a number of users and allows them to access and search its sensitive data stored on the cloud, and each user is allowed to update the data with his unique key (secret key).to achieve dynamic update and search parallel has been adopted on the Keyword-Red-Black(KRB) tree Which does not contain the inverted index [2] constraints that were used in the previous schemes. They were also used the Pseudorandom functions and hash functions which are more efficient than the Bilinear Map in encoding, searching and decoding the inputs

Next, in 2016. O. Dunkelman and L. Keliher [10] developed a devised a novel DSSE scheme with the lowest information leakage, the fastest updates, the smallest client storage, the lowest server storage for large file-keyword pairings, and the simplest design and implementation of all the previous schemes[5][11][4].Data structure used by their scheme (a bit matrix supported by two hash tables) enables efficient yet secure search and update operations, which allows it to attain these desirable qualities. They verify the security of their system and illustrate and They attain a high level of update security with their scheme and that it can be applied to a large number of file-keyword combinations even with a straightforward hardware implementation.

K. Kurosawa et al.[12] They first prove that the UC security and the stand-alone security of dynamic SSE schemes are not equivalent. Thus, one can concentrate on the more straightforward privacy and dependability concept while still achieving high UC security. They then demonstrate a more effective UC secure dynamic SSE approach for the identical model as that of [13] by using XOR-MAC. instead of the RSA accumulator to authenticate the index table. The RSA accumulator is used to authenticate only the encrypted data. The good incrementality attribute of XOR-MAC is used to effectively authenticate the new index table when the client adds a new file. The client also keeps track of the number of files. This enables us to avoid timing the index table. The suggested approach is more effective than [13] because XOR-MAC may be implemented using AES.

The follow them. in 2017, I. Miers and P. Mohassel, [14] provably secure Dynamic Searchable Symmetric Encryption (DSSE) system with a notable decrease in IO cost compared to other efforts when used for email or other highly dynamic message corpora was conceived, built, and evaluated. to using able to reduce the total IO usage by 99% by employing a hybrid approach in which updates are made to a dynamic ORAM-like index and then evicted to a chunked index typically used for static SE, and they are able to reduce the upfront costs of search by 94% by creating a dynamic index that does not protect read privacy. Their approach is still more expensive than non-encrypted search, and deploying for email ultimately comes down to a cost-benefit comparison between the importance of maintaining user privacy and the operating costs.but support single keyword search

T. Kabir and M. A. Adnan. [15] They have devised a scheme that allows cloud servers to execute dynamic operations on data, such as insert, update, and delete, without first decrypting the data. Therefore, by carrying out those duties without decryption, their system not only ensures dynamic operations on data but also offers a secure technique. Modern techniques enable data users to retrieve the data, re-encrypt it in accordance with the new policy, and then transfer it once more to the cloud. However, their proposed approach reduces the pressure placed on data owners to undertake dynamic operations, therefore saving this substantial computational overhead. The kNN technique is used in conjunction with the safe and popular $TF \times IDF$ model to build the index and create the query.They have utilized a tree index structure. but also prevents the exchange of encrypted data, minimizing the computation required of data owners by making full use of previously encrypted data under earlier access rules. The widely used method and the vector space model Inverse document frequency (IDF) and term frequency (TF) are used to build indexes and create queries. Additionally, they created a flexible tree-based index structure that can reach sub-linear search times. This kNN algorithm guarantees accurate similarity calculation

K. S. Kim et al.[16] they developed and put into action the advanced secure DSSE scheme. Through the dual dictionary, a new data structure that combines the forward index for effective update and the inverted index to provide preferred search time, The prior methods did not support the actual deletion of data, which increased storage space requirements, calculation complexity, and claimed inefficiency. This new structure, however, allows data deletion candidly and in real-time. Additionally, their system offers forward security by using fresh keys unrelated to the earlier search tokens to encrypt the freshly added data. Their plan is quite effective in practice for both searches and

updates in dynamic contexts, as demonstrated by the comparison with Sophos.

R. Bost et al.[17] They presented the concept of backward privacy for searchable encryption for the first time and provided many techniques for both forward and backward privacy, each with different efficiency trade-offs. They heavily rely on primitives like restricted pseudorandom functions and puncturable encryption scheme in their creations. By restricting the opponent from evaluating functions on certain inputs or decrypting particular ciphertexts, these sophisticated cryptographic primitives enable a fine-grained control of the adversary's capabilities. Depending on how much metadata about the entered and deleted entries leaks, they established three types of backward privacy, each with a progressively less level of protection:

1.BP-I (Backward privacy with insertion pattern): leaks the documents currently matching w, when they were inserted, and the total number of updates on w.

2.BP-II (Backward privacy with update pattern): leaks the documents currently matching w, when they were inserted, and when all the updates on w happened (but not their content).

3. BP-III (Weak backward privacy): leaks the documents currently matching w, when they were inserted, when all the updates on w happened, and which deletion update canceled which insertion update

and follow them. In 2018 .M. Etemad et al. [18] They introduced a new dynamic SSE technique that provides forward privacy by exchanging the keys exposed to the server for each search. Their system is efficient, parallelizable, and outperforms the best existing forward-privacy schemes, while achieving competitive performance with dynamic schemes lacking forward privacy.

M. S. Niaz and G. Saake.[19] They proposed a DSSE that ensures forward privacy and uses space more efficiently. If a file is deleted, the redundant data nodes in subsequent searches are removed from the secure index. The space reclamation function does not call for complete secure index regeneration. They partially meet the requirements for backward privacy because of the space reclaiming feature.

S. K. Ocansey et al.[20] The multiusers query was supported by their scheme. Moreover, their scheme effectively solved the majority of security holes linked to the disclosure of sensitive information in DSSE schemes. Their method exceeds previous DSSE methods in terms of effectiveness and efficiency, according to simulations and security analyses. Their DSSE with Forward Privacy scheme offers a productive, private, and trustworthy SE scheme. They put out an effective DFP approach that involves creating and periodically storing IoT aggregated files on the cloud. By keeping a rising counter for each keyword at an IoT gateway, their strategy combines forward privacy with sublinear search efficiency. In order to expand their system to a multiuser environment, they proposed an effective technique that gives data owners (DO)-controlled searching enforcement for data users (DUs). This multiuser setting property is implemented on the DO side by the Bloom filter (BF).

T. Hoang et al.[21] In contrast to the use of generic ORAM, they created a series of Oblivious Distributed DSSE schemes known as ODSE that allow oblivious access on the encrypted index with a high level of security and enhanced performance. Particularly, in actual network conditions, ODSE techniques are 3–57 times faster than using the most advanced generic ORAMs on encrypted dictionary index. The information-theoretic security with robustness against malicious servers is one of the desirable security guarantees offered by one of the ODSE techniques that has been proposed. These characteristics are attained by taking use of some special traits of searchable encryption and encrypted index, which enables them to simultaneously harness the processing and communication efficiency of multi-server PIR and Write-Only ORAM. They fully deployed ODSE and ran numerous tests to evaluate how well our suggested methods performed in a genuine cloud environment.

J. Chamani et al [22] Three new backward and forward private SSE schemes were introduced. Mitra is unquestionably the quickest such scheme now in use, beating even those with larger leakage. The search time for their other two structures, Orion and Horus, is now nearly linear to the number of documents in the database that include the keyword w, or n w. The search time in all prior works that achieve backward privacy is (aw), or at least linear in the total number of updates (including deletions) associated to w; in practice, aw can be arbitrarily bigger than n w. Both Orion and Horus do this, but their trade-offs between leakage and search efficiency are different. their work still has many unresolved issues, such as investigating out whether they can develop a system without using ORAM that has a quasi-linear search time and non-trivial communication (known to be possible for schemes that are only forward private). Another approach would be to develop a non-interactive, quasi-optimal search time scheme,

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

or an optimal search time one (which looks difficult for deletion-supporting constructs). In view of potential deletion-specific attacks, it would be interesting to revisit the backward privacy definitions of [7] and assess how much information they leak in practical applications.

In 2019.T. Hoang et al.[23] they developed that[21] by exploiting the multi-cloud infrastructure, They developed [21] a full Oblivious Distributed DSSE (ODSE) framework that allows oblivious search and updates on the encrypted index with high security and enhanced performance over the use of generic ORAM by leveraging the multi-cloud architecture. Their system includes a number of ODSE schemes, each with varying degrees of performance and security required by various sorts of real-world applications. In the presence of a malevolent adversary, ODSE provided desirable security guarantees such as information-theoretic security and resiliency. They completely constructed the ODSE framework and tested its performance in a real-world cloud setting. Under real network conditions, their testing revealed that ODSE techniques are 3-57 quicker than employing generic ORAMs on a DSSE encrypted index.

C. Zuo et al.[24] they proposed two DSSE schemes supporting range queries. The second can achieve backward privacy but can only handle a small number of documents while the first is forward-private and supports a big number of documents. To solve the issue of file-injection attacks[25], where the attacker can undermine the confidentiality of a client query by adding a small number of brand-new documents to the encrypted database, and content leak of deleted documents. The first DSSE build employs their Binary Tree in conjunction with Bost et al [17] framework to achieve forward privacy, However, it incurs a significant storage overhead in the client as well as a significant communication cost between the client and the server. They proposed the second DSSE design with range queries using the Paillier cryptosystem and bit string representation to achieve backward privacy. They employed the fixed update token in this structure to decrease client and server storage at the expense of forward privacy. Furthermore, it cannot support a big number of documents. Despite the fact that the second DSSE structure cannot support a huge number of documents, it can still be highly beneficial in some situations.

P. Rizomiliotis and S. Gritzalis [26] They presented a brand-new DSSE scheme that provides first-line design and accomplishes this level of security with a regular and minimal number of communication roundtrips. They presented a new scheme namely Mara. which provides forward and backward privacy and is considered the most efficient scheme that achieves a level of security and can be proven to be safe considering that the server is honest but fake, that is, it reveals confidential information, but it implements the algorithms correctly and achieves this scheme The privacy has a fixed number of round trips, meaning four messages are exchanged at most. It has a very good asymptotic complexity and they implemented it in practice and evaluated its performance. In addition, it is a dynamic grooved that supports adding and deleting documents in addition to updating document/keyword pairs and uses a scheme Mara two data structures to maintain the search index. an oblivious data structure and an encrypted multi- map (EMM). Since Oram is somewhat impractical, they have introduced a new improvement in the data structure, they called SORAM which is a very simple and effective data structure and it is a simplified version of Path.ORAM where SORAM that allows us to limit the number of back-and-forth communications

Q. Liu et al.[27]They proposed the Verifiable Dynamic Encryption with Ranked Search (VDERS) technique, which enables the user to perform verifiable and efficient updates and top-K searches on ciphertexts, as well as on-demand information retrieval in cloud computing. Their fundamental concept is to build a verifiable matrix to record ranking information and encode it using an RSA accumulator. In addition, a ranking inverted index is constructed from a group of documents to support quick top-K searches and updates. Two constructions, $VDERS^0$ and VDERS*, are offered for efficient top-K searches. $VDERS^0$ is a fundamental construction that enables verifiable document insertion operations, whereas VDERS* is an advanced construction that not only supports efficient deletion processes but also reduces communication costs without outsourcing the verifiable matrix.

C. Zuo et al.[28] They proposed an effective DSSE scheme called FB-DSSE that achieves forward privacy and stronger backward privacy (referred to as Type-I- backward privacy) with only one roundtrip (without taking real file retrieval into account). this scheme is Based on a bitmap index and a straightforward symmetric encryption with homomorphic addition, Later, they expand it to include multi-blocks (named MB-FB-DSSE)

In 2020, I. Demertzis et al.[29] They proposed three schemes to overcome the limitations that were limiting the previous dynamic

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

schemes, namely, forward and backward privacy, and process counters for each unique keyword. It achieves permanent storage on the part of the client They showed three new schemes with constant permanent client storage and better search performance, both asymptotically and experimentally, than previous works. Also, their two schemes, SDa and SDd, eliminate the need for blind accesses during searches and reduce the number of round-trips that are needed. The biggest problem with SDa is that it costs O(logN) to update, which is a lot. This means that some updates are cheap and others are very expensive. This solves the problem above by giving a de-amortized cost for updates. They have shown that for small percentages of deletions, both SDa and SDd perform better (both asymptotically and in practice) than the previous state-of-the-art MITRA. But the main problem with all of these plans is that the search time depends on documents that have already been deleted, which can lead to O(N) search overheads in the worst case. So, for delete-heavy query workloads, they proposed with QOS, which is the most efficient DSE with quasi-optimal search time. It is an improvement of orders of magnitude over the performance of previous quasi-optimal DSE schemes. In many practical situations, they thought that SDd would be a better choice than QOS. However, QOS has been a step towards optimal search performance because it only adds a logarithmic amount of time to the optimal search time. Experiments show that QOS is a better choice when the percentage of deletions is more than 80% (a hypothetical delete-intensive workload).

K. Salmani and K. Barker.[30] They presented the concept of Full Forward Privacy (FFP). establish and define the concept of "full forward privacy" (FFP). The idea behind FFP is that the cloud cannot link a new document to previous queries over time (not only immediately after adding a new document), ensuring the privacy of newly uploaded documents even after the client issues additional queries. This requirement also ensures that the DSSE scheme is immune to search pattern attacks. They built an effective full forward private DSSE scheme with search and update (add and delete) functionality. Furthermore, they proposed a parallelizable DSSE method that achieved FFP by concealing the search pattern with non-deterministic and one-time use search tokens. Their scheme supported private search and private update operations with non-deterministic and one-time use search tokens. Their low-cost scheme allows for both updates and searches.

X. Wanshan et al.[31] They proposed the DESSE encryption scheme, which is based on forward secure searchable encryption. DESSE employed a straightforward encryption scheme. As a result, while encrypting plaintext of the same length, DESSE encrypts shorter ciphertext, increasing encryption and decryption efficiency and conserving space.They presented a dynamic deletion strategy based on a single list during the searchable encryption dynamic deletion procedure. One retrieval can efficiently filter the deleted data by using reverse order retrieval. Scheme DESSE has a lower time complexity than scheme FSSE[32], which searches first and then merges. Furthermore, only the "ind" and "op" are preserved in the FSSE index; hence, in the scenario of many inserts and deletions of the same file, FSSE cannot properly identify the file's latest state; DESSE fixes this problem by introducing the field of update times.

Y. Li et al.[33] they proposed a solution The update approach suffers from high connection and calculation costs, and the server reverts all matched files without relevance rating. They proposed TS-RDSE, which stands for Two-Server Ranked DSE. They combine orthogonal vector and efficient homomorphic encryption cryptosystems to provide a vector-level dynamic secure index that flexibly allows efficient dynamic updating operations such as file deletion and insertion. Additionally, they developed a secure sorting protocol based on the widely-used tf-idf weighting formula and the addition property of partial homomorphic encryption, which achieves accurate ordering for search results while protecting the privacy of relevance scores, in order to rank search results by relevance without decryption..

H.Li et al.[34] They presented two Secure and Efficient Dynamic Searchable Symmetric Encryption (SEDSSE) methods for medical cloud data so that a patient can remotely outsource her medical data to a cloud server and only authorized doctors can access the data due to the sensitivity of medical data. Using the secure k-Nearest Neighbor (kNN) and Attribute-Based Encryption (ABE) techniques, they propose a dynamic searchable symmetric encryption scheme that can achieve two important security features, namely forward privacy and backward privacy, which are extremely difficult to achieve in the field of dynamic searchable symmetric encryption. Then, they presented an improved strategy to address the widespread key sharing issue in kNN-based searchable encryption schemes. Their schemes are superior than existing suggestions in terms of storage, search, and update difficulty.

C.Zuo et al.[35] A safe DSSE system typically requires forward and backward privacy to

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

avoid leakage abuse attacks, however the existing forward and backward private DSSE techniques either only support single keyword queries or require more client-server interactions. First, they present a novel leakage function for range inquiries that is more complex than the one for single-keyword queries. They presented a forward and backward private DSSE (called FBDSSE-RQ) for range queries that required only one roundtrip. In other words, their approach is more efficient since it does not require re-encryption of the matching files for each search. In addition, they refine the construction of the binary tree described in [24]. Instead of the order of node insertion, the names of nodes are taken from their leaf nodes [24]. In addition, they established a novel Type- R reverse privacy concept for their range queries. Due to the binary tree data structure, their range query to update a file with value v leaks the number of keywords that have been modified. From the security and experimental evaluations, they are able to determine that their suggested approach meets the declared security goals and is efficient.

I.Kramer et al.[8] They created a framework for a searchable cipher called (searchect), which is used to enhance applications and search through encrypted data in a non-protocol fashion and could open the stage for more widespread and easy adoption of privacy-enhancing technology. They talked about (DSSE) schemes. Furthermore, forward secure techniques that assure additional security features are supported, allowing for effective updates of the encrypted index. Both are add-on schemes because to the lack of efficient backward secure techniques. Forward secure systems can be supplemented with basic deletion handling, which uses the first byte to indicate whether the document identification is added or deleted. If client performance is critical, the DynRH solution is chosen over the Sophos one. When DynRH augments their existing cloud storage application with their framework, their search queries will cost significantly more bandwidth and a little extra effort. To prevent costly requests, a workaround could be to use the keyword counters in the client state to anticipate the relevance of the phrase in advance. They compare the performance features of two existing forward secure methods, DynRH and Sophos, and find that DynRH surpasses Sophos in terms of efficiency, execution time, update protocol, and search.

J.Blömer et al.[36] They presented a searchable dynamic encryption scheme for a set of dynamic documents shared by several users. Their scheme satisfies the forward and backward privacy, and their contributions assumed a multi-authority attribute-based encryption scheme that secures against adaptive intrusion; this allows for precise control of access to the search results for multiple users and includes access control operations like adding documents to a group of documents, modifying them, or changing individual documents.

J.Chen et al.[37] They presented a newly developed and implemented dynamic symmetric searchable encryption (DSSE) system., a new index structure inspired by the linked list was utilized to provide forward securityI in instead of ORAM .It caters both forward privacy and document deletion during an update. Compared to scheme [37], their approach drastically reduces the computing complexity of the client. In addition, the in the index structure can be used directly to overcome the deletion problem that it does not support [37] and can also be used to solve comparable flaws in other schemes. Their data pool is simpler than its dual dictionary in comparison to [16]. In their design, the client's storage burden is drastically decreased without sacrificing search efficiency. In addition, they are not required to re-encrypt the search results in the encrypted database after each query. They achieve forward security that does not require the server to destroy the key for re-encryption, resulting in great deletion efficiency.

X. Zhang et al.[38] they proposed a DSSE for multiuser with Forward and Backward Security (FBM-DSSE). More particularly, the proposed scheme uses symmetric encryption to improve the efficiency of file encryption and update, a keyed pseudorandom function to conceal the correspondence between files and indexes, primitives of pseudorandom functions (PRF), and the homomorphic message authenticator (HMAC) to create the inverted index and update the search token. to make updating search tokens as files are modified more effective. Additionally, their design can be expanded to multifunctional search and offers verifiability. The goal of future research is to make the data owner and server's calculation and communication processes simpler. Additionally, access control will be combined with file search and fine-grained user management to achieve this.

In 2021.H.Liu et al .[39] they proposed using the obliviously shuffled incidence matrix DSSE (OSM-DSSE) to prevent statistical attacks on DSSE caused by leaks in access and search patterns that restrict the protection of user privacy and restrictions that make the use of ORAM in cloud environments challenging owing to sufficient connection load and the inability to conceal the search pattern. Simply conceal the access

pattern.They suggested it In order to gain unauthorized access to the encrypted data. In particular, a shuffling algorithm using Paillier encryption to solve the problem of access pattern leakage is combined with the 1-out-of-n obliviously transfer (OT) protocol and local differential privacy to obfuscate the search targets. This algorithm can shuffle the data in the incidence matrix to change the access path. The OSM DSSE approach additionally delivers excellent security, effective searches, and decreased storage costs. This scheme not only provides adaptive security against malicious assaults by adversaries but also entirely conceals search and access patterns that used differential privacy based on the random answer to conceal the response length. The OSM-DSSE executes at a pace that is roughly 3–4 times faster than existing methods.

C. Zuo et al.[40] Based on the OXT [41]framework, they provided two DSSE schemes (SDSSE-CQ, SDSSE-CQ-S) with different levels of backward privacy for conjunctive queries (called Type-O and Type-O). The first scheme, known as SDSSE-CQ, allows non-interactive deletion and enhances the forward privacy and Type-O backward privacy. They introduced the second DSSE scheme (called SDSSE-CQ-S) with a higher level of backward privacy Type-O to further limit leakages.As a result, there are less interactions between the server and the client because their schemes do not require sending the deleted files to the server.

K. He et al.[42] They focused on secure DSSE scheme with constant storage cost for clients. Due to the fact that the storage cost on the client side grows linearly with the number of keywords in the database, big keyword sets result in unaffordable storage costs. Their framework is enhanced by fish-bone chain, which ties all keywords to a single state, a revolutionary two-level structure consisting of Logical Keyword Index Chain (LoKIC), the first level of which is a logical structure of search tokens for a keyword. With this structure, they can lower the storage cost on the client side, and with Document Index Chain (DIC), they can optimize the computation cost and reduce the rebuilding overhead. While the storage costs on the client side are constant in both CLOSE-F and CLOSE-FB, which are significantly lower than existing systems, the storage costs on the server side are variable.

Z. Wu et al.[43] They proposed the UI-SE, the first DSE system to achieve single-round-trip interactivity, and near-zero client storage. The OU-tree data structure used in UI-SE, provides oblivious data updates without exposing any access patterns. The OU-tree structure has the benefit of supporting backward-privacy changes with insertion-pattern-hiding. Updates to current keyword-identifier pairs in the index were the study's main focus. Traditional data structures, such as the fish-bone structure [41], can be used with the OU tree to allow different types of data queries and update non-existing pairings.

Z. Li et al.[44] To address the present DSSE schemes, which offer forward and backward privacy secure search but cannot recover the deleted keyword logically or physically, secure forward and backward keyword search with flexible keyword shielding in the multiuser context has been proposed. The FB-AKS scheme Retrievable shielding is accomplished by the use of one-way trapdoor permutation and puncturable encryption, which inhibits file injection assaults and deletion leaks to the cloud server. In comparison to the current forward and backward privacy, FB-AKS obtained keyword authorization flexibility (e.g., keyword shielding, keyword un-shielding). They improved the query efficiency and storage space of FB-AKS.

S.F. Sun et al.[45] They proposed a forward and backward-private DSSE scheme based on the Bloom filter and a multi-puncturable pseudorandom function. Then they demonstrated the first non-interactive (BP -I) backward-private DSSE [17]. They began by introducing a new cryptographic primitive called Symmetric Revocable Encryption (SRE) and proposing a modular architecture based on some concise cryptographic primitives. They presented their DSSE system based on the suggested SRE, implemented it in real networks, and evaluated its practicability and scalability.

C. Xu et al.[46] They proposed a novel Multiple Clouds (BDSSE-MC) scheme for Blockchain-based DSSE. The system enables the data owner to create encrypted local file indexes and, using a smart contract, combines the local indexes into a global index. The smart contract using the global index also does the search operation. The attacker in this scheme simply has access to the number of clouds and files, but do not the original files or search results. To guarantee the fairness and privacy of the plan, smart contracts were used to carry out the protocol and search process.

Y. Zhao .[47] they proposed to expand the current concept of the volume-hiding leakage function into the context of dynamics and offered effective VH-DSSE and VH-DSSE ^{k} constructions. There was a non-negligible accuracy fault in VH-DSSE. to address the drawback of VH-DSSE. They

International Journal of Advances in Engineering and Management (IJAEM)
Volume 4, Issue 9 Sep. 2022, pp: 116-128 www.ijaem.net ISSN: 2395-5252

proposed the VH-DSSE ^{k} multi-copy structure, which enhances accuracy by parallel repetition. The strongest definitions of backward privacy are met by both VH-DSSE and VH-DSSE ^{k}.

A. Pradesh et al.[48] They proposed a novel Dynamic Searchable Symmetric Encryption (DSSE) framework referred to as Incidence Matrix (IM)-DSSE, which achieved a high level of privacy, efficient search/update, and reduced client storage in actual cloud deployments on real cloud settings. They use an incidence matrix and two hash tables to generate an encrypted index on which both search and update operations may be executed with minimal information loss. IM- DSSE concurrently achieved forward-privacy, backward-privacy, and size- obliviousness. In addition, they developed a number of alternative DSSE frameworks, each of which offers different tradeoffs and is suited to various cloud applications and SaaS infrastructures.

M. R. Asghar and S. D. Galbraith.[49] They proposed a protects privacy Multi-cloud-based DSSE scheme for relational Databases t (P-McDb). P-McDb provides minimum leakage, which secures not only the privacy of queries and records, but also the search, intersection, and size patterns. Moreover, P-McDb ensured the database's forward and reverse privacy. P-McDb might therefore withstand existing leakage-based assaults, such as active file/record injection attacks.

C. Chen et al. [50] The verified DSE with ranked search VDERS scheme [27]enables users to verify the accuracy of search results and update outsourced data., however, a recently proposed VDERS approach fails to meet forward privacy since there are two linkages between the prior search token and the appended document. Therefore, they did design an enhanced VDERSc system to accomplish forward privacy. On the basis of VDERS, they proposed VDERSc, a technique that ensures forward privacy. The fundamental idea is to generate a fresh address and search token by adding an additional counter. All of these newly added nodes will be placed into a secure index at the new location, preventing the CSP from deciding which keywords to include. To generate a witness for a search query in VDERS, the CSP must use the RSA accumulator to compute information about additional keywords. In their suggested system, the CSP is only required to calculate information about the search query's matching keyword. In their study, they severed the two connections by incorporating counters and an update buffer.Therefore, VDERSc outperforms VDERS in the proof generating phase.

A. J. Prakash and B. L. Elizabeth.[51] for retrieval of encrypted documents, they proposed PINDEX, a novel private multi-linked dynamic index, in the DSSE. With the use of a secret orthogonal vector and probabilistic homomorphic encryption, this idea generates a secure dynamic index. Support for adding and removing keywords or documents without having to recreate the externally encrypted index using a hash table that holds the inner product of all the rows connected by orthogonal vectors. and provided Through forward privacy Trapdoor or token creation and the key used to build the index are both identical to random functions. This is due to PINDEX's usage of secret orthogonal vectors and probabilistic homomorphic encryption. As a result, the server is unable to discover data by means of searches or access patterns derived from earlier inquiries. This proposed is efficient and parallelizable, allowing multi-keyword search and update operations to operate independently over p processors thanks to a private multi-linked index structure.

J. Gharehchamani et al.[52] They proposed a formal definition of security as well as suitable forward and backward privacy. They created the first forward-backward-private and provably secure DMUSSE schemes and explained how to alter them to obtain outcome verifiability. recent tests demonstrate the superiority of their systems over earlier ones that do not support updates and/or exhibit cross-user leakage due to their extremely low practical overheads.

In 2022. Q.Liu et al.[53] They developed a SE scheme That enables users to outsource ciphertext with search possibility. Although several SE schemes intending to provide high efficiency and proved security have been proposed, classic SE schemes have additionally been subjected to a variety of attacks, causing sensitive data to leak among ciphertexts and queries. In order to prevent the exposure of information that can be linked between the cipher texts and queries, forward and backward-private SE techniques have been devised.They presented a novel backward privacy downgrade attack on the existing SE schemes. To demonstrate the effectiveness of their attack, they demonstrate that MITRA's backward privacy [29] can be downgraded from BP-II to BP-III.

A.Wu et al.[54] They presented One data user can search on encrypted databases using the forward secure searchable encryption (FSSE) scheme while being protected from a file injection attack. By transitioning from the single-user environment to the multi-user scenario, the data consumption can be further enhanced. When a data

owner exchanges data with numerous data users, there are a few difficulties that must be taken into account. First, it's not possible to fully trust the public cloud server because it could be dishonest and return false or incomplete data. Second, authorized users have the option to exchange their private keys for payment. Modern SE scheme only take into account a portion of the following desirable characteristics: the results' verifiability, their resistance to file injection attacks, their traceability, and their ability to revoke the access of malicious users who misuse their private keys in a multi-user environment. they first suggest providing traceable and verifiable multi-user FSSE, which accomplishes the aforementioned features, based on these motivations. Additionally, they perform the security proof to show that their scheme can adhere to security standards.

Cong Zuo et al.[55] they first provided a new leaking function that is more complex than the one for single keyword searches. Also included in their proposal was a practical forward and backward private DSSE system that made use of a more advanced binary tree data structure DSSE allows searches and updates over encrypted databases, recently drawn a lot of interest. A secure DSSE scheme typically involves forward and backward privacy to fend off leakage abuse attack. However, the forward and backward private DSSE techniques now in use either support only single keyword searches or necessitate greater client-server communication. For range queries.

.P. Xu et al.[56] The robustness of DSSE was first defined by them. A robust DSSE scheme can maintain the same correctness and security whether the client repeatedly adds or deletes the same keyword/file-identifier pair or whether the client intentionally deletes a keyword/file-identifier pair that doesn't exist. In response, they added additional timestamps to the duplicate update requests to the original definition of backward security. In contrast, since the previous definition made the implicit assumption that repeated update queries do not occur, just one timestamp was defined. They first create a fundamental DSSE scheme called SEED in order to comprehend their final DSSE scheme. SEED is functional, but not robust, and is BP-III backward secure.With the exception of having the same storage cost as the majority of earlier studies, SEED has the least amount of complexity in terms of computation and communication expenses. SEED is interested in attaining. operations, such as modular exponentiation or bilinear mapping. They defined an instance of the key-updatable PRF cryptographic primitive based on an early PRF scheme in order to convert SEED to their final DSSE scheme. They create the ROSE strong DSSE scheme. Under adaptive attacks, ROSE is BP-III forward secure and BP-III backward secure. With the exception of search performance, it is nearly as complex as SEED in terms of computing and communication costs

VI. CONCLUTION

In cloud computing, DSSE allows for data adding, removing or modification a file after data outsourcing. forward and backward privacy are two important securities of DSSE, most DSSE are insecure due that suffer various attacks and information leakage of sensitive data, such as the search, access, and size patterns. In this paper, we reviewed different dynamic searchable encryption schemes in cloud computing. The main objectives of the DSSE schemes are secure search, efficiency security forward and backward privacy, and efficiency query, build index, most of these objectives were addressed in the proposed scheme In this review, we presented dynamic searchable encryption by considering encryption techniques used and algorithms, and performance evaluation in recent work from 2015 to 2022, Which addressed many of the challenges faced by researchers in the development and design DSSE.

REFERENCES

[1] K. Huang, X. Dong, Z. Cao, and J. Shen, “Dynamic searchable symmetric encryption schemes with forward and backward security,” in IOP Conference Series: Materials Science and Engineering, 2020, vol. 715, no. 1, doi: 10.1088/1757-899X/715/1/012062.

[2] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: Improved definitions and efficient constructions,” J. Comput. Secur., vol. 19, no. 5, pp. 895–934, 2011, doi: 10.3233/JCS-2011-0426.

[3] D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy, 2000, pp. 44–55, doi: 10.1109/secpri.2000.848445.

[4] S. Kamara, C. Papamanthou, and T. Roeder, “Dynamic searchable symmetric encryption,” in Proceedings of the ACM Conference on Computer and Communications Security, 2012, pp. 965–976, doi: 10.1145/2382196.2382298.

[5] D. Cash et al., "Dynamic Searchable Encryption in Very-Large Databases: Data Structures and Implementation," 2014.

[6] E.Stefanov et al, "Practical Dynamic Searchable Encryption with Small Leakage Storing private files in the cloud."

[7] B. Salim Sabah Bulbul Supervisor and A. I. Ayad Abdulsada, "Dynamic Searchable Encryption Scheme," 2021.

[8] I. Kramer, S. Schmidt, M. Koschuch, and M. Tausig, "Experimental evaluation of forward secure dynamic symmetric searchable encryption using the searchitect framework," in IoTBDS 2020 - Proceedings of the 5th International Conference on Internet of Things, Big Data and Security, 2020, pp. 25–35, doi: 10.5220/0009337000250035.

[9] W. Luo, Y. Chen, and Y. Zhou, "Dynamic searchable encryption with multi-user private search for cloud computing," in Proceedings - 15th IEEE International Conference on Computer and Information Technology, CIT 2015, 14th IEEE International Conference on Ubiquitous Computing and Communications, IUCC 2015, 13th IEEE International Conference on Dependable, Autonomic and Se, Oct. 2015, pp. 176–182, doi: 10.1109/CIT/IUCC/DASC/PICOM.2015.359.

[10] O. Dunkelman and L. Keliher, "Preface," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9566, p. v, 2016, doi: 10.1007/978-3-319-31301-6.

[11] S. Kamara and C. Papamanthou, "Parallel and dynamic searchable symmetric encryption," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2013, vol. 7859 LNCS, pp. 258–274, doi: 10.1007/978-3-642-39884-1_22.

[12] K. Kurosawa, K. Sasaki, K. Ohta, and K. Yoneyama, "UC-Secure dynamic searchable symmetric encryption scheme," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9836, pp. 73–90, 2016, doi: 10.1007/978-3-319-44524-3_5.

[13] K. Kurosawa and Y. Ohtaki, "How to update documents verifiably in searchable symmetric encryption," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8257 LNCS, pp. 309–328, 2013, doi: 10.1007/978-3-319-02937-5_17.

[14] I. Miers and P. Mohassel, "IO-DSSE: Scaling Dynamic Searchable Encryption to Millions of Indexes By Improving Locality," 2017, doi: 10.14722/ndss.2017.23394.

[15] Association for Computing Machinery, Institute of Electrical and Electronics Engineers. Bangladesh Section, Bangladesh University of Engineering and Technology. ACM Chapter, Bangladesh University of Engineering and Technology. Department of Computer Science and Engineering, and Institute of Electrical and Electronics Engineers, Proceedings of 2017 4th International Conference on Networking, Systems and Security (NSysS) : 18-20 December, 2017, Dhaka, Bangladesh. .

[16] K. S. Kim, M. Kim, D. Lee, J. H. Park, and W. H. Kim, "Forward secure dynamic searchable symmetric encryption with e‡icient updates," in Proceedings of the ACM Conference on Computer and Communications Security, Oct. 2017, pp. 1449–1463, doi: 10.1145/3133956.3133970.

[17] R. Bost, B. Minaud, and O. Ohrimenko, "Forward and backward private searchable encryption from constrained cryptographic primitives," in Proceedings of the ACM Conference on Computer and Communications Security, Oct. 2017, pp. 1465–1482, doi: 10.1145/3133956.3133980.

[18] M. Etemad, A. Küpçü, C. Papamanthou, and D. Evans, "Efficient Dynamic Searchable Encryption with Forward Privacy," Proc. Priv. Enhancing Technol., vol. 2018, no. 1, pp. 5–20, Sep. 2018, doi: 10.1515/popets-2018-0002.

[19] M. S. Niaz and G. Saake, "Forward secure searchable symmetric encryption," in 2017 12th International Conference for Internet Technology and Secured Transactions, ICITST 2017, 2018, pp. 49–54, doi: 10.23919/ICITST.2017.8356345.

[20] S. K. Ocansey, W. Ametepe, X. W. Li, and C. Wang, "Dynamic searchable encryption with privacy protection for cloud computing," Int. J. Commun. Syst., vol. 31, no. 1, Jan. 2018, doi: 10.1002/dac.3403.

[21] T. Hoang, A. A. Yavuz, F. B. Durak, and J. Guajardo, "Oblivious dynamic searchable encryption on distributed cloud systems," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10980 LNCS, pp. 113–

130, 2018, doi: 10.1007/978-3-319-95729-6_8.

[22] J. G. Chamani, D. Papadopoulos, C. Papamanthou, and R. Jalili, "New constructions for forward and backward private symmetric searchable encryption," Proc. ACM Conf. Comput. Commun. Secur., pp. 1038–1055, 2018, doi: 10.1145/3243734.3243833.

[23] T. Hoang, A. A. Yavuz, F. B. Durak, and J. Guajardo, "A multi-server oblivious dynamic searchable encryption framework," J. Comput. Secur., vol. 27, no. 6, pp. 649–676, 2019, doi: 10.3233/JCS-191300.

[24] C. Zuo, S.-F. Sun, J. K. Liu, J. Shao, and J. Pieprzyk, "Dynamic Searchable Symmetric Encryption Schemes Supporting Range Queries with Forward/Backward Privacy," May 2019, [Online]. Available: http://arxiv.org/abs/1905.08561.

[25] D. Cash, P. Grubbs, J. Perry, and T. Ristenpart, "Leakage-abuse attacks against searchable encryption," Proc. ACM Conf. Comput. Commun. Secur., vol. 2015-Octob, pp. 668–679, 2015, doi: 10.1145/2810103.2813700.

[26] P. Rizomiliotis and S. Gritzalis, "Simple forward and backward private searchable symmetric encryption schemes with constant number of roundtrips," Proc. ACM Conf. Comput. Commun. Secur., pp. 141–152, 2019, doi: 10.1145/3338466.3358921.

[27] Q. Liu, Y. Tian, J. Wu, T. Peng, and G. Wang, "Enabling Verifiable and Dynamic Ranked Search over Outsourced Data," IEEE Trans. Serv. Comput., vol. 15, no. 1, pp. 69–82, 2019, doi: 10.1109/TSC.2019.2922177.

[28] C. Zuo, S. F. Sun, J. K. Liu, J. Shao, and J. Pieprzyk, "Dynamic Searchable Symmetric Encryption with Forward and Stronger Backward Privacy," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11736 LNCS, pp. 283–303, 2019, doi: 10.1007/978-3-030-29962-0_14.

[29] I. Demertzis, J. G. Chamani, D. Papadopoulos, and C. Papamanthou, "Dynamic Searchable Encryption with Small Client Storage," Feb. 2020, doi: 10.14722/ndss.2020.24423.

[30] K. Salmani and K. Barker, "Dynamic Searchable Symmetric Encryption with Full Forward Privacy," 2020 IEEE 5th Int. Conf. Signal Image Process. ICSIP 2020, pp. 985–995, 2020, doi: 10.1109/ICSIP49896.2020.9339338.

[31] X. Wanshan, Z. Jianbiao, and Y. Yuan, "DESSE: A Dynamic Efficient Forward Searchable Encryption Scheme," IEEE Access, vol. 8, pp. 144480–144488, 2020, doi: 10.1109/ACCESS.2020.3012975.

[32] Y. Wei, S. Lv, X. Guo, Z. Liu, Y. Huang, and B. Li, "FSSE: Forward secure searchable encryption with keyed-block chains," Inf. Sci. (Ny)., vol. 500, pp. 113–126, 2019, doi: 10.1016/j.ins.2019.05.059.

[33] Y. Li, F. Zhou, Z. Xu, and Y. Ge, "An Efficient Two-Server Ranked Dynamic Searchable Encryption Scheme," IEEE Access, vol. 8, pp. 86328–86344, 2020, doi: 10.1109/ACCESS.2020.2992773.

[34] Y. W. Ti, C. F. Wu, C. M. Yu, and S. Y. Kuo, "Benchmarking Dynamic Searchable Symmetric Encryption Scheme for Cloud-Internet of Things Applications," IEEE Access, vol. 8, pp. 1715–1732, 2020, doi: 10.1109/ACCESS.2019.2961971.

[35] C. Zuo, S. Sun, J. K. Liu, J. Shao, J. Pieprzyk, and L. Xu, "Forward and Backward Private DSSE for Range Queries," IEEE Trans. Dependable Secur. Comput., vol. XX, no. XX, pp. 1–1, 2020, doi: 10.1109/tdsc.2020.2994377.

[36] J. Blömer and N. Löken, "Dynamic Searchable Encryption with Access Control," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2020, vol. 12056 LNCS, pp. 308–324, doi: 10.1007/978-3-030-45371-8_19.

[37] J. Chen, Z. Cao, J. Shen, X. Dong, and X. Wang, "Forward secure dynamic searchable symmetric encryption with lighter storage," in PervasiveHealth: Pervasive Computing Technologies for Healthcare, 2020, pp. 24–30, doi: 10.1145/3377644.3377666.

[38] X. Zhang, Y. Su, and J. Qin, "A Dynamic Searchable Symmetric Encryption Scheme for Multiuser with Forward and Backward Security," Secur. Commun. Networks, vol. 2020, 2020, doi: 10.1155/2020/8893016.

[39] H. Liu, X. Li, E. Guo, Y. Xiao, and T. Li, "OSM-DSSE: A Searchable Encryption Scheme with Hidden Search Patterns and Access Pattern," 2021, doi: 10.21203/rs.3.rs-253711/v1.

[40] C. Zuo, S. Lai, X. Yuan, J. K. Liu, J. Shao, and H. Wang, "Searchable Encryption for

Conjunctive Queries with Extended Forward and Backward Privacy."

[41] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M. C. Roşu, and M. Steiner, "Highly-scalable searchable symmetric encryption with support for Boolean queries," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8042 LNCS, no. PART 1, pp. 353–373, 2013, doi: 10.1007/978-3-642-40041-4_20.

[42] K. He, J. Chen, Q. Zhou, R. Du, and Y. Xiang, "Secure Dynamic Searchable Symmetric Encryption with Constant Client Storage Cost," IEEE Trans. Inf. Forensics Secur., vol. 16, pp. 1538–1549, 2021, doi: 10.1109/TIFS.2020.3033412.

[43] Z. Wu, J. Wang, S. Member, and K. L. Fellow, "Update-Sensitive Structured Encryption with Backward Privacy."

[44] Z. Li, J. Ma, Y. Miao, X. Liu, and K. K. R. Choo, "Forward and backward secure keyword search with flexible keyword shielding," Inf. Sci. (Ny)., vol. 576, pp. 507–521, Oct. 2021, doi: 10.1016/j.ins.2021.06.048.

[45] S.-F. Sun et al., "Practical Non-Interactive Searchable Encryption with Forward and Backward Privacy," Ndss, no. February, 2021, [Online]. Available: https://dx.doi.org/10.14722/ndss.2021.24162.

[46] C. Xu, L. Yu, L. Zhu, and C. Zhang, "A blockchain-based dynamic searchable symmetric encryption scheme under multiple clouds," Peer-to-Peer Netw. Appl., 2021, doi: 10.1007/s12083-021-01202-6.

[47] Y. Zhao, H. Wang, and K. Y. Lam, "Volume-Hiding Dynamic Searchable Symmetric Encryption with Forward and Backward Privacy," pp. 1–35, 2021.

[48] A. Pradesh and A. Pradesh, "ISSN NO : 0377-9254 PageNo : 208 Vol 12 , Issue 12 , Dec / 2021 ISSN NO : 0377-9254 PageNo : 209," vol. 12, no. 12, pp. 208–219, 2021.

[49] M. R. Asghar and S. D. Galbraith, "Privacy-preserving Dynamic Symmetric Searchable," vol. 24, no. 3, 2021.

[50] C. M. Chen, Z. Tie, E. K. Wang, M. K. Khan, S. Kumar, and S. Kumari, "Verifiable dynamic ranked search with forward privacy over encrypted cloud data," Peer-to-Peer Netw. Appl., vol. 14, no. 5, pp. 2977–2991, Sep. 2021, doi: 10.1007/s12083-021-01132-3.

[51] A. J. Prakash and B. L. Elizabeth, "PINDEX: Private multi-linked index for encrypted document retrieval," PLoS One, vol. 16, no. 8 August, pp. 1–22, 2021, doi: 10.1371/journal.pone.0256223.

[52] J. Gharehchamani, Y. Wang, D. Papadopoulos, M. Zhang, and R. Jalili, "Multi-User Dynamic Searchable Symmetric Encryption with Corrupted Participants," IEEE Trans. Dependable Secur. Comput., vol. 5971, no. c, pp. 1–16, 2021, doi: 10.1109/TDSC.2021.3127546.

[53] M. Yoo, H. Yoon, C. Hahn, D. Koo, and J. Hur, "Downgrading Backward Privacy of Searchable Encryption," Int. Conf. Inf. Netw., vol. 2022-Janua, pp. 324–328, 2022, doi: 10.1109/ICOIN53446.2022.9687109.

[54] A. Wu, A. Yang, W. Luo, and W. Jinghang, "Enabling Traceable and Verifiable Multi-user Forward Secure Searchable Encryption in Hybrid Cloud," IEEE Trans. Cloud Comput., vol. 14, no. 8, 2022, doi: 10.1109/TCC.2022.3170362.

[55] C. Zuo, S. F. Sun, J. K. Liu, J. Shao, J. Pieprzyk, and L. Xu, "Forward and Backward Private DSSE for Range Queries," IEEE Trans. Dependable Secur. Comput., vol. 19, no. 1, pp. 328–338, 2022, doi: 10.1109/TDSC.2020.2994377.

[56] P. Xu et al., "ROSE: Robust Searchable Encryption with Forward and Backward Security," IEEE Trans. Inf. Forensics Secur., vol. 17, pp. 1115–1130, 2022, doi: 10.1109/TIFS.2022.3155977.