A SYSTEMATIC REVIEW OF CRYPTDB: IMPLEMENTATION, CHALLENGES, AND FUTURE OPPORTUNITIES

Hana Yousuf, The British University in Dubai
Said A. Salloum, University of Salford, University of Sharjah
Ahmad Aburayya, Dubai Health Authority
Mostafa Al-Emran, The British University in Dubai
Khaled Shaalan, The British University in Dubai

ABSTRACT

In the case of compromised databases or interested database managers, CryptDB has been built for validated and realistic protection. CryptDB operates through encrypted data while executing SQL queries. The key concept of the SQL-aware encryption technique is to map SQL operations to encryption methods, adjustable query-driven encryption which facilitates CryptDB to modify the encryption level of data depending on user queries and to alter the data through layered encryption levels in an efficient manner. The systematic literature review in this paper shows that there is ongoing research regarding the implementation of CryptDB in new applications such as cloud computing and management information systems. Experiments are being conducted to improve the encryption schemes and layers to avoid data leakage when CryptDB is applied in dynamic applications. Further, there are studies on alternative query-processing systems to improve the performance and throughput. However, CryptDB is found to be the only practical approach to process the queries for encrypted data.

Keywords: Encrypted Database, CryptDB, Onion Schema, Layered Encryption, Database Security, Query Processing, Proxy, Cryptographic Systems

INTRODUCTION

Theft of confidential personal details is a major concern. Databases are specifically appealing to attackers as they comprise a large volume of personal information. When the public or businesses store their confidential data in a Database Management System (DBMS), they will rely on the resolute equipment and applications and trust the degree of physical security of data centres and the database administrators. Therefore, an attacker with access to any of these components would be able to compromise the entire database as evident from reports on system hacks and fraudulent activities.

CryptDB was built for tested and practical security in the case of corrupted databases or motivated database managers. CryptDB utilizes authenticated details when running SQL queries. The SQL-aware encryption technology primarily includes the task of mapping SQL operations with the support of encryption techniques, which enables CryptDB to adjust the encrypted data level according to user requests and to effectively modify the data by layered encryption rates. CryptDB simply allows the application to perform queries submitted by users and guarantees full anonymity given the mixture of users' queries. The database server completely evaluates the encrypted data queries and returns the results back to the client to be decrypted for penultimate purposes. CryptDB ensures that clients do not process queries and client-side applications operate unchanged.

Citation Information: Yousuf, H., Salloum, S.A., Aburayya, A., Al-Emran, M., & Shaalan, K. (2021). A systematic review of CRYPTDB: Implementation, challenges, and future opportunities.. *Journal of Management Information and Decision Sciences, 24*(S1), 1-16.

This study aims to conduct a systematic review on the technology of CryptDB. The focus of this research is on its implementation in different real-time applications, challenges that are encountered due to the encryption techniques used, and future opportunities for CryptDB in comparison with other newly emerging techniques.

LITERATURE REVIEW

A literature review was performed as part of this study to understand the concept of CryptDB, its evolution, practical uses, disadvantages, challenges in implementation, and future outlook.

Background

DBMSs operating over encrypted data are gaining significant commercial interest. CryptDB, designed by Popa, Redfield, Zeldovich & Balakrishnan (2011) is one such notable system that supports a variety of SQL queries over encrypted databases. It is a practical system offering a throughput loss of only 26% as compared to MySQL.

Encrypted Database Management System

DBMSs are an especially enticing option for attackers, because they contain vast amounts of private knowledge as well. If today a customer or business needs confidential data in a DBMS, it is critical that the hardware and software of the server are uncompromising, that the data centre itself is under physical security and that the DBAs are trustworthy. An attacker, who has access to either of those forms, can otherwise jeopardise the whole database, as reported in several recent data theft reports (Dayioglu, Kiraz, Birinci & Akin, 2013).

The architecture and deployment of DBMS, which functions on encrypted files, poses three major challenges. The first is that a wide variety of SQL queries for authenticated data is to be supported using a DBMS to conduct calculations on encrypted data to operate SQL queries. This is unlike simple encryption, where each row in a database is encrypted with a single key.

The second challenge is related to the privacy issue in a DBMS and the systems design. The DBMS should be capable of processing encrypted data while performing operations such as aggregation, selection and join to process SQL queries related to row. At the same time, an expert may have knowledge about the interconnectivity between rows and the decryption involved. Therefore, data security is to be ensured in such a manner that the server-side processing matches the need to reduce the data exposed to the server (Mallaiah & Ramachandram, 2014).

The third problem is regarding the functional implementation of an authenticated DBMS. To work efficiently, an authenticated DBMS will enforce minimal overhead efficiency on the application, thus eliminating the SQL database update. To easily operate an encrypted DBMS, an ideal system would not require modifications to the existing database systems either. Any encryption technique should be able to incorporate the decades of engineering and optimisation development to ensure that DBMS servers operate over a wide range of technologies and client applications without making changes to them. CryptDB was developed to address these challenges.

CryptDB

There is a growing commercial interest for DBMSs that operate through encrypted data (Popa, Zeldovich & Balakrishnan, 2011). CryptDB is one of the exceptional tools that allow a number of authenticated data SQL queries. This is a functional system that utilises a variety of encryption schemes and a new basic authentication to enable a SQL connector. This modern basic encoding system acts as an adaptive communication tool and helps construct tokens that refer to any two columns in the database to calculate the link even if they are encrypted. CryptDB is a tool for the protection analysis of adaptable joining schemes; however, what forms of potential adversarial actions it includes are not well known. In fact, the CryptDB join operator is transitive, signifying that it has the ability to reveal critical details about the information stored in an encrypted database.

In 2011, Popa et al. proposed a software-only server solution called CryptDB to use the practical deployment of a trustworthy proxy server through a more relaxed security system, *i.e.*, confidentiality, with no integrity. A major advantage of CryptDB is that storage systems remain secure without disrupting the organisation's business processes. The device is highly coupled and thoroughly validated with MySQL. CryptDB expertly encompasses DET, OPE, homorphical additive encryption and Song's idea to allow encrypted text field searches.

Popa, et al., ensured that the database is safe as well as the proxy server is trusted excluding certain vulnerabilities such as frequency analysis, order relation and queries hit. To be precise, the computers that hold the data base management program and its managers are not trusted but trust the client and the proxy server.

CryptDB Configurations

To enable realistic applications, CryptDB is configured to protect MySQL databases. CryptDB allows developers to create encrypted databases with an estimated overhead output of 21%–26% on MySQL. In general, only a few code lines are enough to run CryptDB in applications. CryptDB aims to overcome the weaknesses of existing solutions that are either too slow or lack the required confidentiality. Throughout the usual framework of data-backed systems of DBMS servers and independent client servers, CryptDB includes a proxy service and several other components (Alves & Aranha, 2018).

There are three approaches that CryptDB uses to solve the issues related to database security.

- SQL-aware encryption approach: SQL queries have a well-known framework, consisting of operators including order comparisons, equivalent control and additional aggregates, such as total and table joins. This is the technique used by SQL-aware encryption. CryptDB then converts the requests into a type that enables the DBMS to conduct encrypted data using cryptographic methodologies.
- The customisable query-based encryption: This solves the issue of other cryptographic schemes that leaks more data than necessary. Only because they are still necessary, to change the queries, the encryptions are required, which reduces the vulnerability of data.
- Linking user passwords by cryptographic keys: The third approach protects users who are not signed into a network which links encryption keys with user passwords, enabling users with access privileges to decrypt data.

Queries over Encrypted Data

To protect records, any interactions that can be read from the database are covered by modifying the standard schema and saving it in the proxy of CryptDB (Akin & Sunar, 2014). Even authenticated are tables and column titles. The encryption of columns depends on the column data and the DBMS type of queries to be performed. Based on the form of data in a column, there are six encryption methods applied in CryptDB.

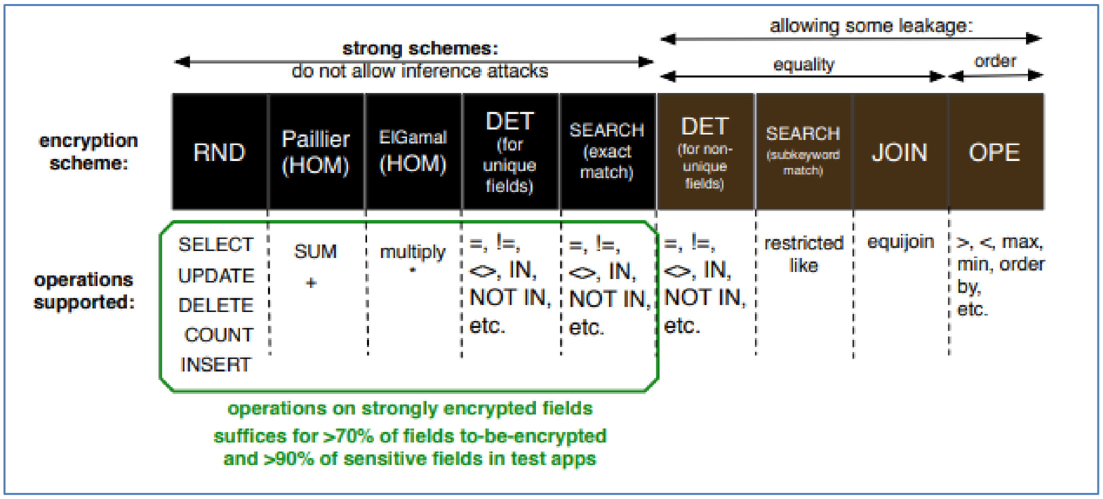
allowing some leakage:
strong schemes:
do not allow inference attacks
equality
order
encryption scheme:
RND
Paillier (HOM)
ElGamal (HOM)
DET (for unique fields)
SEARCH (exact match)
DET (for non-unique fields)
SEARCH (subkeyword match)
JOIN
OPE
operations supported:
SELECT UPDATE DELETE COUNT INSERT
SUM +
multiply *
=, !=, <>, IN, NOT IN, etc.
=, !=, <>, IN, NOT IN, etc.
=, !=, <>, IN, NOT IN, etc.
restricted like
equijoin
>, <, max, min, order by, etc.
operations on strongly encrypted fields
suffices for >70% of fields to-be-encrypted
and >90% of sensitive fields in test apps

FIGURE 1
ENCRYPTION SCHEMES IN CRYPTDB

*Source: Popa, Zeldovich & Balakrishnan (2015)

Random (RND): A randomly created Initial Vector (IV) is used to construct a ciphertext from the column word. A strong encryption is provided by RND, which is suitable for sensitive data handling. Therefore, it enables queries to be operated that require computation; for example, ORDER BY, Min and SUM queries.

Deterministic (DET): The defense offered by the leakage is weaker due to a shortage of the same ciphertext being generated for the same text. It is a pseudo-random permutation.

Order-Preserving Encryption (OPE): It retains the ciphertext sequence to remain in plaintext. For instance, if a < b, OPEK(a) < OPEK(b), which is applicable for a key K. It is weaker than DET.

Homomorphic (HOM): For any data which requires computation, homomorphic is useful. It makes it easier to create complicated mathematical equations, as plaintext might be produced.

JOIN and OPE-JOIN: As separate DET keys are used, joins are used to enter columns and mask the connections between columns. Equality and order checking are accomplished through joins.

Word checks (SEARCH): The encrypted query reaches the DBMS with the encryption keys. It runs successfully with a few user-defined functions (UDFs). The data that it returns to the proxy is decrypted and sent to the application.

With the encryption key, the authenticated application enters the DBMS. A few UDFs operate successfully. The data returned to the proxy is then decrypted and submitted to the client.

Features of CryptDB

The main features of cryptDB are listed below (Wasankar & Deorankar, 2018).

- They use strong encryption methods.

- There are several query types applicable, since various kinds of encryption techniques and application modifications are used.
- It is fast. It stems from the analysis of the actual phpBB, HotCRP and gradation trials.
- A complex technique for transmitting specific data sets to various users is supported by layered encryption schema.

Limiations of CryptDB

CryptDB has some theoretical limitations as identified below (Wasankar & Deorankar, 2018).

- High vulnerability is implicit in user data.
- Cryptographic key protection is overhead for the whole framework.
- Intensive computations are used to encrypt a query.
- It does not suffice to use a single encryption method; therefore, it increases the overhead.
- CryptDB often leaks data, helping attackers learn the interface, which will eventually allow them to interfere with user data.

CryptDB Threat Coverage

CryptDB was designed and considered to survive two types of threats (Devi & Chakravarthy, 2017).

Threat 1 concerns undermining the database server security. CryptDB watches against a suspicious database manager or external intruder in this risk. It follows a passive intrusion paradigm and fails assaults that jeopardise the server credibility.

Threat 2 contains attacks directed at the whole system, including the device, the client and the database server. It is believed that the intrusion would continue for a limited time and that only people accessing the data would be infected by information during that duration, as the existing files in the database are secured.

Moreover, columns classified as "sensitive" are guaranteed by CryptDB. In other terms, the database application administrator will decide which fields are important, and CryptDB holds them protected through powerful encoding schemes. In this scenario, the encryption exposes no data element information other than their lengths in these columns. The security of the database does not require inference attacks, even though the intruder has side details on plaintext material in the database for an intruder that steals an authenticated document (Popa, Zeldovich & Balakrishnan, 2015).

METHODOLOGY

A comprehensive literature review should establish the study technique theory. This paper reflects the approach used to perform the intentional literary examination utilising the systemic analysis outlined by Kitchenham, Turner, Bailey & Linkman (2009). The explanation behind this comprehensive analysis is that there is no formal assessment based on the usage and drawbacks of neural networks series to series. In turn, this technique allows one to compile, test, assess and investigate different implementations, challenges faced and future opportunities of CryptDB.

Research Questions

The research questions for this systematic literature review are listed below:

1. What are the different approaches in which CryptDB is implemented for encrypted databases?
2. What are the applications of CryptDB?
3. What are the limitations and challenges experienced while implementing cryptDB?
4. Are there any alternative approaches available to overcome the shortcomings of CryptDB?
5. Which country has contributed highly to the research studies on CryptDB?

Table 1
RESEARCH QUESTIONS & EXPLANATIONS

S.no	Question	Explanation
1	What are the different approaches in which CryptDB is implemented for encrypted databases?	This query aims to provide a better view of all CryptDB implementations. The best practicable solution is to apply the model and then allow it to be calculated depending on the potential results for each application.
2	What are the applications of CryptDB?	This query offers an overview into CryptDB's growth. In fact, it should highlight the shortcomings in recognising additional research in the ground and the possible and least troublesome method for the progress of IT software.
3	What are the limitations and challenges experienced while implementing CryptDB?	This topic discusses the value of the paradigm and whether the method has managed to mitigate some of the challenges inherent with its growth. In fact, this helps highlight those problems that have not yet been overcome and new ones that the current model has created.
4	Are there any alternative approaches available to overcome the shortcomings of cryptDB?	This problem stresses the advantages and drawbacks of CryptDB against other solutions and the correct way to propose a framework for the application of this concept.
5	Which country has contributed highly to the research studies on CryptDB?	This helps to determine the geographical regions where there is an increased interest towards the research and development of CryptDB.

Research Strategy

The search for literature covered the 2015–2020 period and included university research papers and technology journals. Textbooks were used to collect theoretical information, and Google Scholar was used to search for published articles. This research focuses on understanding the implementation, challenges involved as well as the future potential of CryptDB. The research is based on the ability of the CryptDB such as overhead, throughput and changes to queries required. The literature search used key words such as CryptDB implementation, challenges, potential and encrypted DBMS. The search results are split into three types of papers as follows:

1. Papers related to CryptDB
2. Papers related to CryptDB security
3. Papers related to similar cryptographic systems

A detailed analysis of corresponding records, peer-reviewed articles and periodicals was the theory of study policy for this paper. Most arXiv articles are included in academic journals, Google Scholar, ScienceDirect and IEEE journal articles. The keywords from the research hypothesis were selected in the search for the most important records by means of relevant titles and revised articles and research papers. Boolean methodology was modified to merge relevant search words to find, for the intent of this paper, the most meaningful objects examined by the peers. For instance, similar words or plural expressions and other viewpoints were found while using the search terms. Additionally, 1260 documents were released during the initial stage of the search. Such articles were refined, and the final articles included in this report were chosen and checked in compliance with the PRISMA framework for systematic reviews (Liberati, et al., 2009).

Table 2
STRING SEARCH

Search Database	String	Count of Results
Google Scholar	"CryptDB"+"implementation"	1060
	"CryptDB"+"Challenges"	707
	"CryptDB" and "future"	903
arXiv	"CryptDB"	1950
ScienceDirect	"CryptDB"	33
IEEE	"CryptDB" and "implementation"	15
Springer	"CryptDB" and "future"	3

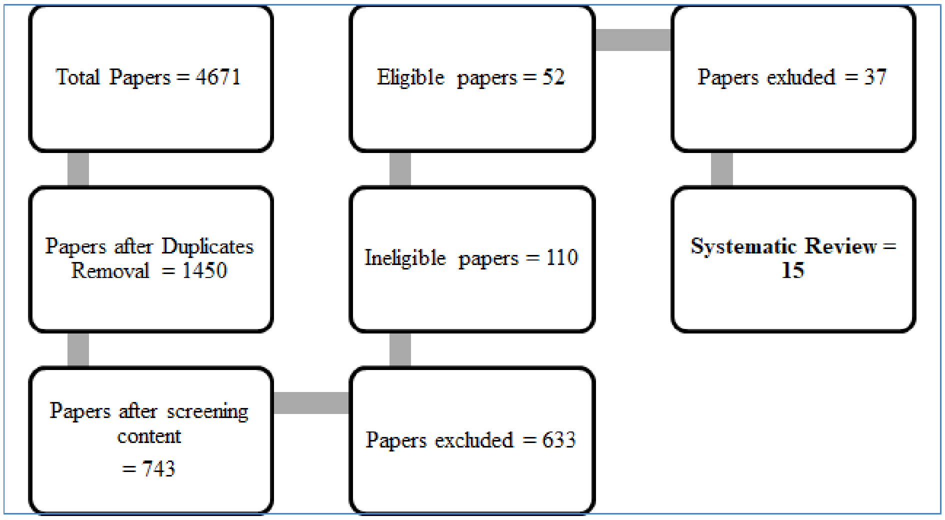

FIGURE 2
SELECTION OF RESEARCH PAPERS FOR SYSTEMATIC REVIEW

Inclusion and Exclusion Criteria

The requirements for inclusion and exclusion were chosen to determine the right text for this paper's scope. Accordingly, the articles that adopted the analysis approach regarding the purpose of study requirements were provided (Table 3).

Table 3
INCLUSION AND EXCLUSION CRITERIA

Inclusion Criteria	Exclusion Criteria
Peer-reviewed journal articles, research papers and books sections	Thesis publications
Content of the articles should discuss the implementation, challenges and future potential of CryptDB	Papers on specialised function of CryptDB

Citation Information: Yousuf, H., Salloum, S.A., Aburayya, A., Al-Emran, M., & Shaalan, K. (2021). A systematic review of CRYPTDB: Implementation, challenges, and future opportunities.. *Journal of Management Information and Decision Sciences, 24*(S1), 1-16.

English language papers	Non-English research papers
Time of publication between 2015 and 2020	Time of publication before 2015

The work was removed from scientific articles and did not follow the aforementioned standards. To restrict the outcome, the original review was rendered for all 4671 documents obtained. Such reports were related to pre-selection requirements, namely the improvement of engineering and computer science study areas, which restricted the findings to no longer than five years. The analysed numbers were limited to 52 articles that were assessed using the aforementioned criterion. Figure 2 reflects the process for pre-selection and selection requirements. This cycle enabled the comprehensive analysis of CryptDB to include 15 papers.

Quality Assessment

One of the most common and crucial aspects in all systematic assessments is consistency assurance. This comprehensive analysis consists of six questions for the evaluation of quality control. As shown in the following table, 15 papers were selected. An answer of “yes” to the consistency evaluation is indicated by 1; “no” is indicated by 0; and "weak" is indicated by 0.5. This measure is based on a study performed by AlEmran, Mezhuyev, Kamaludin & Shaalan (2018).

- Is the purpose of research explicitly stated?
- Are the details accurately and concisely presented?
- Can the research clarify its analysis sufficiently?
- Will the findings of the analysis lead to recognising the reality and difficulties of CryptDB?
- Are the conclusions well-defined?
- Are the results rational and succinct according to the document flow?

The limitation of this systematic review is that the selection of the research papers was based on the understanding and perception of the researcher about the area under study as well as the exclusion and inclusion criteria set to select the research papers.

RESULTS

The four research questions were focused on the testing technique mentioned in the previous section. Depending on the input for the issue as shown, each paper was categorised and evaluated.

Classification and Analysis of the Studies

A grading method focused on responses to the answers to the research questions was used in the analysis of the 15 research papers included in the systematic examination. Markings were made if the paper's main focus was on a specific category. Some articles, for example, quickly addressed how CryptDB is to be used in different programmes, problems with the new security process and developments and compared them with alternatives. That analysis was further examined in depth, and the findings of this thorough research are presented. The country-specific information is given on the basis of the papers selected for the review (Table 4).

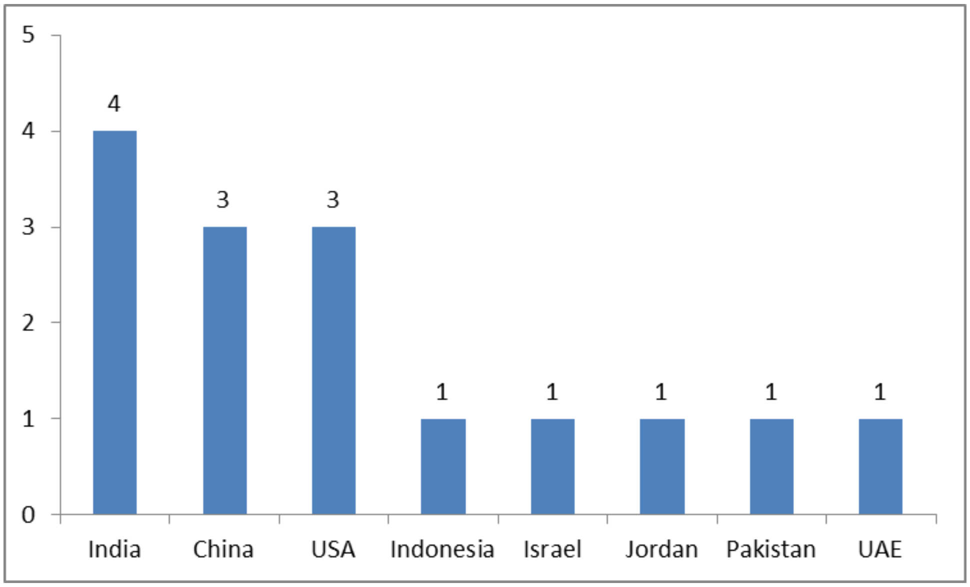

FIGURE 3
SELECTED PUBLICATION BY COUNTRY

Table 4
ANALYSIS OF THE STUDIES

S.no	Source	Purpose	Country	Database	Findings
S1	Mironov, Segev, and Shahaf, (2017)	To strengthen the existing CryptDB joins	Israel	Springer	A scheme improves protection by the provision of a non-transitive joiner when encoding CryptDB from one group to four group elements on the grounds of linear expectation and raising its period from one group exponentiation to the one dependent on linear assumptions on the one machine category exponentiation of four bilinear maps.
S2	Foltz and Simpson, (2018)	The goal for this work was to determine the feasibility for a full-scale implementation on a real Oracle Enterprise Resource Planning (ERP) system.	United States	Google Scholar	It includes external functionality such as management processes, views and controls on multi-user usage. The research demonstrates that such new functions can be technically introduced using authenticated data and enforced in such a way that the functionality of the ERP program system does not require adjustments. The total latency specifications for requests and the computing resources to run on encrypted data are in one order, and the amount of unencrypted data is a tiny component. Such findings suggest that an Oracle ERP can be run on authenticated data.
S3	Liu, Yang, Wang, Xiang, and Dai,	To support different operators in SQL queries	China and	Google Scholar	A simple and safe-free FHOPE coding enables cloud servers to

	(2018)	over encrypted data, multiple encryption schemes need to be combined and adjusted to work together. Moreover, repeated encryptions will reduce the eficiency of execution.	Australia		execute complicated SQL queries to include various operators over encrypted data without repeated encryption (such as add, multiplication, order comparison and equality controls). Such operators are interoperable files, signifying that complicated SQL queries can be merged. The planned FHOPE system is evaluated in safety and efficiency. The trial results indicate that the system costs less overhead on computation and coordination than current methods. This is suitable for massive, complicated SQL batch queries over encrypted data for use in cloud.
S4	Wasankar and Deorankar, (2018)	The database is managed by the cloud provider; however, database items are encrypted with keys that are only known by the data owner. SQL queries run over the encrypted database using a collection of operations such as equality checks and order comparisons.	India	Google Scholar	CryptDB uses encryption systems which permit ciphertexts to make these comparisons. CryptDB is a weak attacker paradigm, as it believes that a reliable cloud-based application server and proxy exists. Nonetheless, CryptDB plays a fascinating role regardingthe trade-off between flexibility and cloud providers' secrecy.
S5	Yousuf and Salloum, (2020)	To make the vectorised data secure, we must apply a security method, which will be CryptDB.	United Arab Emirates	Google Scholar	Through evaluating the responses from software security engineers, both word2vec and CryptDB work remarkably. Word2vec is a powerful solution to vectorisation, and CryptDB offers an accurate and safe database.
S6	Devi and Chakravarthy, (2017)	Data being stored in the cloud is susceptible to intruders, where one of the intruders can be the database administrator. To protect the critical data stored in the cloud from the database administrator, CryptDB can be used.	India	Google Scholar	For Relational Cloud, CryptDB is applicable. Relational Cloud is a database-as-a-service. It utilises security methods to encrypt, save, and decrypt data to retrieve the data through CryptDB.
S7	Burkhalter, Hithnawi, Viand, Shafagh, and Ratnasamy, imeCrypt: Encrypted Data Stream Processing at Scale with Cryptographic Access Control, (2020)	A growing number of devices and services collect detailed time series data that is stored in the cloud. Protecting the confidentiality of this vast and continuously generated data is an acute need for several applications in this space. At the same time, the	USA	Google Scholar	TimeCrypt is a system that provides scalable and real-time analytics over large volumes of encrypted time series data. TimeCrypt allows users to define expressive data access and privacy policies and enforces it cryptographically *via* encryption. In TimeCrypt, data is encrypted end-to-end, and authorised parties can only decrypt and

		utility of this data must be preserved by enabling authorised services to securely and selectively access and run analytics.			verify queries within their authorised access scope. TimeCrypt as an alternative for CryptDB is shown.
S8	Yao and Shuai, (2018)	CryptDB is an encrypted database management system proposed by CSAIL Lab of MIT. It can ensure the privacy of private data by executing SQL queries on the encrypted data. However, some encryption algorithms used in CryptDB might be time-consuming. Homomorphic encryption (HOM) is a secure probabilistic encryption scheme, and in CryptDB, the author-implemented HOM is Paillier Cryptosystem, which is a probabilistic asymmetric algorithm for public key cryptography. Due to the complex computation, Paillier has become a bottleneck of the system performance.	China	IEEE Xplore	Chinese Remainder Theorem(CRT) is used in CryptDB to accelerate the process of encryption. The results show that this method improves the performance of CryptDB under certain conditions.
S9	Popa, Zeldovich, and Balakrishnan, Guidelines for Using the CryptDB System Securely, (2015)	To review the guidelines used for the secure implementation of CryptDB	USA	Google Scholar	A summary of the guidelines to ensure that researchers do not incorrectly apply CryptDB securely by the administrators of database applications.
S10	Nasereddin and Darwesh (2020)	To create insert object (to insert new records) to insert data to multi table and multi columns and update object (update records) and delete object (delete records), using Java language, because Java is an object oriented programming language; also, Java has a rich application program interface (API) and powerful development tools such as Eclipse and NetBeans	Jordan	Google Scholar	Building an advanced object of JAVA language to insert, update and delete and one that is able to handle both encrypted and decrypted data without taking care of the encryption or decryption processes. Moreover, this object can be called by the programmer without the need to write the SQL query every time as it is built in the object, insert data to multi tables, providing the ability to use an object without saving the names of the table and the column in the object.
S11	Shahzad, Iqbal, and Bokhari, (2015)	To determine the performance of CryptDB with OpenEMR on different deployment scenarios and varying workloads over the cloud	Pakistan	IEEE Xplore	CryptDB successfully provides the data confidentiality on the database server when deployed on the cloud. It was also found that for a mixed workload, the average performance of the

		and a local testbed			OpenEMR with CryptDB in the cloud remains under two seconds, which makes CryptDB a viable option for providing security to EHR systems deployed in the cloud.
S12	Jiang, Kong, and Xu, (2020)	To determine and improvement scheme for the shortcomings of the original CryptDB system	China	Google Scholar	An improved additive Order-Revealing Encryption (aORE) scheme by combining the Practical Order-Revealing Encryption (P-ORE) and mOPE. The scheme is based on the pseudorandom function and double encryption. Compared with mOPE, it can improve the execution efficiency of the Order-Preserving scheme at the expense of security.
S13	Gor and Jain, (2016)	Data is stored on a third-party server; therefore, security of data is the main concern. There is data that can be stored in an encrypted form, however it is not convenient every time to download the encrypted data, decrypt and modify it and again encrypt the data and upload it on cloud storage.	India	Google Scholar	A solution to data encryption in the cloud layer of encryption on data which secures the data in the layer on encryption algorithm and query which execute on data is also encrypted on encrypted data and get data. Onion encryption technique is one of them.
S14	Prakruti, Dara, and Muralidhara, (2015)	Many companies are skeptical about moving their critical applications to the cloud due to privacy issues associated with outsourcing sensitive data to third-party cloud providers.	India	Google Scholar	This study proposes storage-efficient, SQL-aware encrypted databases that preserve the format of the fields. It shows experimental results of storage improvements in CryptDB using the FNR encryption scheme.
S15	Maisura, (2018)	The data stored in the database are all in encrypted form. CryptDB is a new cryptographic technique, where the system acts as a proxy to protect the communication between the application server and database server, implementing the idea.	Indonesia	Google Scholar	This study looks into more detail about the encryption scheme implemented in CryptDB in two different case studies, using the SEARCH command with the condition given. The result of the study presents how the statement changes with the act of proxy and the encryption scheme implemented here.

Quality Assessment Results

Using the quality assurance questions identified in the previous section, each research paper was reviewed, and the results are plotted as shown in the following Table 5. The papers were selected in a suitable manner to answer the research question and meet the quality control requirements; accordingly, the papers were rated as 0 for low and 1 for high quality.

Table 5
SELECTED PAPER QUALITY ASSURANCE RATING

Paper	QA1	QA2	QA3	QA4	QA5	QA6	Total	Percentage
S1	1	1	1	1	1	0.5	5.5	92%
S2	1	1	0.5	1	1	1	5.5	92%
S3	1	1	1	1	1	1	6	100%
S4	1	1	1	1	1	1	6	100%
S5	1	1	1	0.5	0.5	1	5	83%
S6	1	1	0.5	1	1	1	5.5	92%
S7	1	1	1	0.5	1	1	5.5	92%
S8	1	1	0.5	1	1	1	5.5	92%
S9	1	1	1	0.5	1	0.5	5	83%
S10	1	1	1	1	1	1	6	100%
S11	1	1	1	1	0.5	1	5.5	92%
S12	1	1	0.5	1	0.5	1	5	83%
S13	1	1	1	1	1	1	6	100%
S14	1	1	1	1	1	1	6	100%
S15	1	1	1	0.5	1	0.5	5	83%

As the results show in Table 6, there has been an increase in interest in this topic over the last five years due to security requirements, especially in the aspects of cloud outsourcing and ERP implementation.

Table 6
RESEARCH AREAS OF THE SELECTED ARTICLES

Source	Implementation	Challenges	Future Opportunities	Alternatives
S1	Yes	Yes	Yes	No
S2	Yes	Yes	Yes	No
S3	Yes	Yes	Yes	Yes
S4	Yes	Yes	Yes	No
S5	Yes	Yes	Yes	No
S6	Yes	Yes	Yes	No
S7	Yes	Yes	Yes	Yes
S8	Yes	Yes	Yes	No
S9	Yes	Yes	Yes	No
S10	Yes	Yes	Yes	Yes
S11	Yes	Yes	Yes	No
S12	Yes	Yes	Yes	No
S13	Yes	Yes	Yes	No
S14	Yes	Yes	Yes	No
S15	Yes	Yes	No	No

The main focus of these studies were to enhance the technical aspects to improve the security of the database when applied in real-time complex applications.

Answers to Research Questions

The answers for the research questions are as follows.

RQ1. What are the Different Approaches in which CryptDB is Implemented for Encrypted Databases?

CrypDB is primarily used in encrypted databases. The literature review shows that it is implemented in the database server's side. However, increasing research is being conducted on implementing it on the client's side (Maisura, 2018). This is expected to increase the security of the query processing at both ends of the server.

RQ2. What are the Applications of CryptDB?

Even though CryptDB is used for the security of databases, it is increasingly being recommended as a security mechanism in cloud computing and ERP. The studies by Devi and Chakravarthy (2017); Gor & Jain (2016) as well as Wasankar & Deorankar (2018) show that there are higher future potentials for CryptDB in these applications. The research paper of Foltz and Simpson (2018) and Shahzad, Iqba & Bokhari (2015) show that CryptDB can be integrated with other organisational systems–especially, ERP and management information systems – to ensure the security of information queried from them. It is also found to improve the security in vector data sets (Yousuf & Salloum, 2020).

RQ3. What are the Limitations and Challenges Experienced while Implementing CryptDB?

The current review shows that CryptDB has challenges and limitations regarding the performance when introduced to heavy loaded systems. There are increased chances of informational leakages due to weaker onion layers in it. To overcome this, several researchers have identified problem areas and developed corresponding extensions or upgrades. For instance, the encryption schemes were assessed (Jiang, Kong & Xu, 2020), and extensions were experimented with to improve the security functions (Prakruti, Dara & Muralidhara, 2015; Mironov, Segev & Shahaf, 2017; Yao & Shuai, 2018).

RQ4. Are there any Alternative Approaches Available to Overcome the Shortcomings of CryptDB?

In the selected research papers, other encryption techniques for query processing in databases were assessed against CryptDB as found in TimeCrypt to manage time series data (Burkhalter, Hithnawi, Viand, Shafagh & Ratnasamy, 2020) and using object-oriented language to program query processing (Nasereddin & Darwesh, 2020). Though the experiments denote that the outcomes are higher or on par with CryptDB, the practicality of these alternatives is questionable and further research is required to adress the same.

CONCLUSION

CryptDB is the first practical tool for running most standard queries on encrypted data. It does not make any changes to the DBMS. Although CryptDB shows scalability in implementation across different applications, there are weak links in its layered encryption schema that have to be sorted out. Though there is ongoing research in determining efficient query processing systems to substitute CryptDB, the practical use is still under study. Overall, CryptDB is designed to overcome the weaknesses of current solutions which are either too slow or do not provide the necessary confidentiality.

REFERENCES

Akin, I.H., & Sunar, B. (2014). On the difficulty of securing web applications using CryptDB. *IEEE Xplore*, 745-752.

Alves, P.G., & Aranha, D.F. (2018). A framework for searching encrypted databases. *Journal of Internet Services and Applications, 9,* 1.

Burkhalter, L., Hithnawi, A., Viand, A., Shafagh, H., & Ratnasamy, S. (2020). Ime crypt: Encrypted data stream processing at scale with cryptographic access control. *17th {USENIX} symposium on networked systems design and implementation, 835-850, NSDI.*

Burkhalter, L., Hithnawi, A., Viand, A., Shafagh, H., & Ratnasamy, S. (2020). Time crypt: Encrypted data stream processing at scale with cryptographic access control. *17th {USENIX} Symposium on Networked Systems Design and Implementation (835-850), NSDI.*

Capuyan, D.L., Capuno, R.G., Suson, R., Malabago, N.K., Ermac, E.A., …& Lumantas, B.C. (2021). Adaptation of innovative edge banding trimmer for technology instruction: A university case. *World Journal on Educational Technology, 13*(1), 31-41.

Dayioglu, Z.N., Kiraz, M.S., Birinci, F., & Akin, I.H. (2013). Secure database in cloud computing: CryptDB revisited. *6th International Information Security & Cryptology Conference (pp. 94-104), Ankara: ISC.*

Devi, M., & Chakravarthy, K. (2017). A survey of security aspects of CryptDB in cloud. *International Journal of Engineering Trends and Technology (IJETT),* 273-277.

Foltz, K., & Simpson, W.R. (2018). Extending CryptDB to operate an erp system on encrypted data. *Proceedings of the 20th International Conference on Enterprise Information Systems (ICEIS 2018) (pp. 103-110). SCITEPRESS–Science and Technology Publications.*

Gor, M., & Jain, G. (2016). Survey on cloud database security using onion encryption techniques. *International Institution for Technological Research and Development, 1*(6), 3.

Jiang, X., Kong, X., & Xu, Z. (2020). Research on order-preserving encryption scheme based on CryptDB. *Journal of Physics: Conference Series, 1550*(3), 032106.

Kitchenham, B.O., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering–A systematic literature review. *Information and software technology, 51*(1), 7-15.

Liberati, A., Altman, D.G., Tetzlaff, J., Mulrow, C., Gøtzsche, P.C., Ioannidis, J.P., . . . Moher, D. (2009). The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration. *Journal of clinical epidemiology, 62*(1), e1-e34.

Liu, G., Yang, G., Wang, H., Xiang, Y., & Dai, H. (2018). A novel secure scheme for supporting complex SQL queries over encrypted databases in cloud computing. *Security and Communication Networks,* 15.

Maisura, M. (2018). Analysing client-side encryption implemented in CryptDB. Cyberspace. *Journal of Information Technology Education, 2*(1), 69-83.

Mallaiah, K., & Ramachandram, S. (2014). Applicability of homomorphic encryption and CryptDB in social and business applications: Securing data stored on the third party servers while processing through applications. *International Journal of Computer Applications, 100*(1), 5-19.

Mironov, I., Segev, G., & Shahaf, I. (2017). Strengthening the security of encrypted databases: Non-transitive JOINs. *Theory of Cryptography Conference, 631-661,* Springer.

Nasereddin, H.H., & Darwesh, A.J. (2020). An object oriented programming on encrypted database system (CryptDB). *Journal of Talent Development and Excellence, 12*(1), 5140-5146.

Popa, R.A., Redfield, C.M., Zeldovich, N., & Balakrishnan, H. (2011). CryptDB: Protecting confidentiality with encrypted query processing. *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, 85-100.

Popa, R.A., Zeldovich, N., & Balakrishnan, H. (2011). *CryptDB: A practical encrypted relational DBMS*. MIT-Computer Science and Artificial Intelligence Laboratory.

Popa, R.A., Zeldovich, N., & Balakrishnan, H. (2015). *Guidelines for using the CryptDB system securely*. IACR Cryptol, 979.

Prakruti, C., Dara, S., & Muralidhara, V.N. (2015). Efficient format preserving encrypted databases. *IACR Cryptol*, 219-222.

Shahzad, F., Iqbal, W., & Bokhari, F.S. (2015). On the use of CryptDB for securing electronic health data in the cloud: A performance study. *IEEE Xplore*, 120-125.

Wasankar, N.W., & Deorankar, A. (2018). Secure database in cloud computing: CryptDB overview. *International Journal of Advance Research in Science and Engineering*, *7*(3), 404-408.

Yao, L., & Shuai, X. (2018). Accelerate the paillier cryptosystem in CryptDB by Chinese remainder theorem. *IEEE Xplore*, 74-77.

Yousuf, H., & Salloum, S. (2020). Survey analysis: Enhancing the security of vectorization by using word2vec and CryptDB. *Advances in Science, Technology and Engineering Systems Journal*, *5*(4), 374-380.

Citation Information: Yousuf, H., Salloum, S.A., Aburayya, A., Al-Emran, M., & Shaalan, K. (2021). A systematic review of CRYPTDB: Implementation, challenges, and future opportunities.. *Journal of Management Information and Decision Sciences, 24*(S1), 1-16.