Search, Find and Resolve: Towards a Taxonomy for Searchable Encryption Schemes

Ines Kramer, Silvia Schmidt, Mathias Tausig and Manuel Koschuch[a]

Competence Centre for IT Security, University of Applied Sciences FH Campus Wien, Vienna, Austria

Keywords: Searchable Encryption, Taxonomy, Inverted Index, Forward Index, Full-domain Search.

Abstract: Searchable Encryption (SE) schemes are a promising solution to the problem of outsourcing one's data to a cloud provider in a secure way, while still retaining the ability to search for and easily retrieve specific documents. A multitude of different schemes have been proposed and designed, yet in general they still lack usability/applicability for a specific use case or proper security analysis in order to be widely implemented and used. To address this issue we started a project to determine which SE schemes fit certain use cases - mainly focusing on usability. We examined nearly 400 papers on SE schemes from the last 13 years and extracted categorization domains for SE schemes. Furthermore we took a time-based look at these domains and tried to identify future trends in SE technologies. In this position paper we introduce our methodology and give a short overview of our current work-in-progress.

1 INTRODUCTION AND RELATED WORK

Since Song et al. published their seminal work in 2000 (Song et al., 2000), research on Searchable Encryption (SE) delivered numerous articles and diverse SE schemes. Multiple improvements regarding usability and security were achieved, while also new threat models and attack scenarios were discovered.

In addition to that, papers with various approaches towards establishing an SE taxonomy or respective surveys have been published. The first survey-like article was released by Sedghi et al. (Sedghi et al., 2008) in 2008 and proposed a categorization based on information leakage, i.e. information leakage while writing data vs. information leakage while performing search on data.

In 2011 Boneh et al. (Boneh et al., 2011) provided definitions on *Functional Encryption*, which were divided into *Predicate Encryption* and *Predicate Encryption with Public Key*. The taxonomy by Tang (Tang, 2012) was published the following year. He defined four main SE categories based on the kind of encryption and what kind of search is performed i.e. asymmetric and symmetric encryption - each divided into index-based and full-domain search.

A categorization based on functionalities of Privacy-Assured SE schemes was published by Li et al. in 2013 (M. Li et al., 2013). They used a top-down methodology for designing privacy-assured search schemes: search functionality → information retrieval → data index structure → primitive data operation → cryptographic design → SE primitives.

The most comprehensive survey on SE to date was written by Bösch et al. (Bösch et al., 2014). They mainly differentiate between asymmetric and symmetric encryption. A further categorization by means of single or multi writer(s)/user(s) was introduced, which led to four classes in the form of SE Writer/Reader: SE Single Writer (S)/Multi Reader (M), SE S/S, SE M/M, and SE M/S.

The more recent position paper by Cui et.al. (Cui et al., 2017) analyses 24 schemes according to their leakage patterns and security requirements. Any SE scheme suffers from a form of information leakage to the server, which is described by leakage patterns. The *index/size pattern* refers to the information that can be deduced from the stored ciphertext/index. This includes documents or index size, number of documents or keywords, document lengths and similarity. The *search pattern* reveals the same keyword by comparing the matched records of two queries. By examining the history of result-sets in repeated queries the *access pattern* can be inferred.

Based on these articles we categorized recent SE work by tagging almost 400 papers mainly re-

[a] https://orcid.org/0000-0001-8090-3784

Kramer, I., Schmidt, S., Tausig, M. and Koschuch, M.
Search, Find and Resolve: Towards a Taxonomy for Searchable Encryption Schemes.
DOI: 10.5220/0007753804140421
In *Proceedings of the 4th International Conference on Internet of Things, Big Data and Security (IoTBDS 2019)*, pages 414-421
ISBN: 978-989-758-369-8

trieved from IEEE, IACR, and other (preferably peer-reviewed) online repositories. Our initial motivation was to find a scheme suitable for implementation in an existing cloud system. Soon we realized that we had to start by categorizing the existing schemes and by doing so were able to identify various directions of SE research.

In general, SE research focuses on three aspects (Bösch et al., 2014):

- Efficiency
- Security
- Query expressiveness

Our main focus is set on identifying and implementing a usable and well-performing (by means of runtime and memory) scheme for practical use. With this in mind, Tang's (Tang, 2012) taxonomy lacks categories for practical applications, e.g. number of users with read and write access privileges.

Therefore the approach by Bösch et al. (Bösch et al., 2014) is convincing in terms of determining an SE scheme for practical usage.

SE has to focus on three different trade-off scenarios (Bösch et al., 2014):

- security vs. efficiency
- query expressiveness vs. efficiency
- security vs. query expressiveness

When increasing the security of a scheme, efficiency almost always suffers. The less secure the system is built, the more efficient it usually can perform. Furthermore, SE schemes should offer high expressiveness in search options, which often leads to less secure and/or less efficient systems. It depends on the specific individual application scenario which key figures have to be prioritized: Can the content of the dataset be derived by domain knowledge, and which level of security is required? How big is the underlying dataset, and is there a need for dynamic changes? How often will queries be performed? Thus, elaborating certain criteria regarding SE research (mainly attributes of SE schemes) is indispensable.

In this position paper we want to give some insights into our current work-in-progress. Since we try to cover the majority of work published since SE schemes were initially introduced, this paper currently only discusses our intermediate results up to papers published during 2016. We are aware that this discussion is currently severely lacking works published in the last 2 years (IEEE alone lists almost 170 publications for this period, as much as in 2014 to 2016 combined), but we are currently actively working on finishing our analysis.

1.1 Technical Corpus

We began by dividing the overall 393 articles on SE into 6 topics, see Table 1.

Table 1: Segmentation of the surveyed papers into general topics. Some papers have been assigned to several topics, so the *Ratio* column adds up to more than 100%.

Topic	Ratio
SE Background Information	23.4%
SE Surveys	2.3%
SE Security Analysis & Attacks	4.6%
SE Applications	4.3%
SE Schemes	66.2%
SE Implementations	31.0%

SE Applications represents work on explicit applied usage of SE schemes, whereas *SE Implementations* contains articles which also provide an evaluation of the implemented SE scheme.

Often, implementations focus on testing efficiency (for a variety of metrics) and security for certain SE algorithms of the scheme. Therefore, the majority of implementations do not consider integration into a client/server framework.

Furthermore, the SE schemes were categorized by cryptographic settings ((a)symmetric/hybrid), search settings (full-domain/index-based/hybrid), query expressiveness, and data types. Several articles focused on more than one topic.

91 (23.2%) articles of all 393 collected papers focus on diverse background information for SE, i.e. certain aspects of SE schemes are elaborated in these articles. These aspects are given in Table 2 and give a certain indication of the research trends in underlying primitives.

Table 2: Technical aspects dealt with in the identified SE Background Papers.

Technical Aspects	Ratio
Bilinear Maps	22.0%
Homomorphic Encryption	18.7%
Identity-Based Encryption	13.2%
Private Information Retrieval	9.9%
Attribute-Based Encryption	7.7%
Lattice-Based Systems	4.4%
Functional Encryption	2.2%
Hidden-Vector Encryption	2.2%
Broadcast Encryption	2.2%

The remainder of this position paper is now structured as follows: We give an overview on the different schemes and their corresponding domains we encountered during our research in Section 2, followed by the extraction of an SE-development-timeline in Section 3 in order to find out if we can identify certain trends in the development of SE. We finally conclude in Section 4 by giving a short outlook on our planned future work.

2 MAIN CATEGORIZATION DOMAINS

Throughout our research of 393 articles on SE (published between 2000 and December 2016) we identified 260 different SE schemes and the main domains for categorizing these schemes involved:

- number of users (writer/reader)
- cryptographic primitives
- types of search
- search criteria
- types of data
- security

In the remainder of this Section we will give a short overview of our findings for each domain.

2.1 Number of Users (Read/Write Access Privilege)

Categorizing by means of writer(s) and reader(s) (Bösch et al., 2014) in an SE scheme is of significant importance regarding the practical applicability of such a scheme. Choosing an SE scheme for a given purpose immediately raises questions on access privileges. Who is allowed to *write* (i.e. generate and transmit encrypted data to the server) and who is allowed to *read* (i.e. submit a query by issuing a trapdoor and interpret its results).

In 39 (15%) of 260 papers on SE schemes no information on the number of users, in terms of writer(s) and reader(s) was found. The remaining papers contain

- 72 S/S (27.7%; single writer/single reader) systems,
- 52 S/M (20%; single writer/multiple readers) systems,
- 72 M/S (27.7%; multiple writers/single reader) systems, and
- 34 M/M (13.1%; multiple writers/multiple readers) systems.

All SE schemes with any kind of multiple users (*/M or M/*) are suitable for sharing data. S/S schemes are mainly built for outsourcing data for a single entity. All */M SE settings require some mechanism for key distribution and access control to allow multiuser reading.

2.2 Cryptographic Primitives

When classifying SE schemes regarding their cryptographic primitives, one always looks at the way how these primitives are used in SE related operations (e.g. search queries, index) and not how the actual data is encrypted. So it still possible (and common) for an *asymmetric* scheme to work with *symmetrically* encrypted data.

In general, in a symmetric scheme there exists one secret key. Whoever holds that key can add data to the repository and perform searches on that data.

In an asymmetric setting, the entities in possession of the private key can issue search queries and retrieve data, while possession of the public key allows for data to be added to the repository.

Currently there doesn't seem to be a general consensus of when to call a system *hybrid*, so when giving this classification we followed the one given by the authors in their original publication.

Among the 260 articles on SE schemes we identified

- 117 symmetric schemes (45%),
- 132 asymmetric schemes (50.8%), and
- 19 hybrid schemes (7.3%)

(some papers focused on more than one scheme).

The first symmetric SE scheme (SSE) published in 2000 (Song et al., 2000) represents the S/S setting as described in Subsection 2.1. Asymmetric SE schemes where initially called public key encryption (PKE) schemes. In 2004 PEKS (Public Key Encryption with Keyword Search) was introduced by Boneh et al. (Boneh et al., 2004), originating the current term PEKS for asymmetric settings and M/S SE schemes.

2.3 Types of Search

There are basically two types of search in SE: the full-domain search and the index-based search. Both searches blind the search queries to avoid data leakage (Tang, 2012).

Full-domain search means that the search is performed over all data, while the search in index-based systems is performed on pre-defined keywords gathered in an index. The keywords have to be carefully selected to allow for meaningful queries, but this technique is usually less complex and more flexible in terms of data encryption. Its drawbacks are the decreased flexibility when updating the index, additional memory requirements for storing the index, and the expensive (re-)building of the index.

The index-based search is divided into three subcategories: forward index, inverted index, and hybrid index. In a forward index, each data item is associated with a list of keywords (cf. Table 3). Worst case search complexity is quadratic (for n documents with n keywords each). Yet updating the index (be it by changing keywords or documents) is usually fast and easy.

Table 3: Example of a Forward Index (Tang, 2012).

Data Item	Index
Document 1	black, red, white, chrome, vienna
Document 2	blue, white, copper, gold, memphis
Document 3	yellow, orange, gold, rome, vienna

An inverted index is one single index representing all data (cf. Table 4). It is a further possibility for minimizing search time since it is now linear in the number of keywords. However updating this index is usually a more costly and complex operation. Schemes that allow for efficient updates of the index are called dynamic, while static schemes do not support insert, update and delete operations.

Table 4: Example of an Inverted Index (Tang, 2012).

Data Item	Index
black	Document 1
red	Document 1
chrome	Document 1
vienna	Document 1, Document 3
blue	Document 2
white	Document 1, Document 2
copper	Document 2
memphis	Document 2
yellow	Document 3
orange	Document 3
gold	Document 2, Document 3
rome	Document 3

The 260 articles on SE schemes contain

- 19 full-domain search systems (12.3%; 12 asymmetric, 7 symmetric) and
- 129 index-based search systems (87.7%; 49 asymmetric, 80 symmetric; with 8 hybrid, 13 forward index, 28 inverted index, and the remaining schemes without an explicitly stated index structure).

Further we identified 26 schemes explicitly considered to be dynamic, 20 of them index based. The first dynamic inverted-index approach has been proposed in (Liesdonk et al., 2010).

The remainder of the schemes were not explicitly associated with a certain type of search.

2.4 Search Criteria

Search criteria must not be neglected when evaluating the usability of an SE scheme. Different applications have different requirements in terms of query expressiveness. Early SE schemes offered search setting for queries with a single keyword; today most systems support at the very least multiple keywords.

This categorization also includes the results of a query, e.g. ranked results. In our review, we identified the search criteria given in Table 5.

We additionally encountered the following search criteria in less than five schemes:
comparison, semantic search, subset queries, structured query (SQL), phrase search, and verifiable search.

These low numbers may be caused by the fact that these kinds of query where only introduced near the end of our investigated body of work, e.g. structured query was first published in 2016.

By default, SE search queries are equality queries, thus equality search was not explicitly listed in Table 5.

Boolean queries support boolean operations such as conjunction, disjunction, and negation on keywords. Ranked search delivers the query results ranked by a pre-defined order, for example by relevance to the given keyword (query) (Wang et al., 2010). In systems with range queries (incl. subset queries) the search is performed in a certain range within the data, e.g. the first 100 documents. Most users nowadays are used to fuzzy keyword search (e.g. Google), where even typos and incomplete submitted keywords produce the desired search results. Therefore, fuzzy keyword search is seen as a further step towards usability - like similarity search.

2.5 Types of Data

A further criterion for categorizing SE schemes is the kind of data SE is performed on. By default all the SE schemes are elaborated on simple file servers containing text documents, but in our investigated set there are also

- 26 databases,

Table 5: Query Types supported in the investigated papers.

Query Type	Ratio
Boolean Queries	21.9%
Ranked Results	14.2%
Range Queries	9.2%
Fuzzy Keyword Search	5.4%
Similarity Search	3.1%
Delegated Search	2.3%

- 7 audit logs,
- 5 mobile devices,
- 4 public health records, and
- 2 email settings.

Furthermore, we found geodata, face recognition systems, genomic data, and images.

2.6 Security

To gain a deeper insight into the security constraints of SE schemes we evaluated several leakage abuse attacks. These are divided into two main branches according to their impact.

- Partial plaintext recovery attacks against schemes with fully-revealed occurrence pattern with keyword order (L3) and query-revealed occurrence pattern (L2) as described in (Cash et al., 2015; Naveed et al., 2015).
- Query recovery attacks, which reveal the queried keyword(s). These generally do not depend on the underlying SE scheme. They also work for query revealed occurrence pattern schemes (L1), as used in most of the examined articles.

The first Query recovery attack proposed by (Islam et al., 2012) exploits access pattern leakage. With prior knowledge of the used dataset, the contents of the search queries can be guessed with high accuracy. These attacks are even possible if a scheme is proven to be secure under the standard assumption, which means these attacks are permitted by default.

Cash et al. increased the efficiency of this attack in (Cash et al., 2015). Their count attack is based on the fact that a large fraction of keywords will match against a unique number of documents. Hence, an adversary who knows the plaintext documents simply counts the number of documents returned by each keyword and compares it to the number of documents matched by a query.

An attack based on the search pattern has been presented by Liu et al. (Liu et al., 2013). An adversary who has knowledge of the user's search habit can effectively attack the keywords underlying the query with the help of some publicly available knowledge.

Zhang et al. extended the active known document attacks from (Cash et al., 2015) which rely on the access pattern and named them file injection attacks. The authors propose a non-adaptive and an adaptive version depending on inserting files before or after search queries are made. These powerful attacks work for all dynamic schemes which are not forward secure, even if they provide a low leakage. An adversary controlling the server is able to learn a very high fraction of keywords searched by a client using a relatively small number of injected files. Table 6 gives an overview of the different leakage attacks currently defined in the literature.

Table 6: Summary of the different leakage abuse attacks.

	(Islam et al., 2012)	(Cash et al., 2015)	(Liu et al., 2013)	(Zhang et al., 2016)
Property	n.a.	n.a.	n.a.	dynamic
Leakage Pattern	Access	Access	Search	Access
Required Information	Known plaintext dataset	Partial known plaintext dataset	Publicly known search pattern	Injected files

Due to the power of file injection attacks we extended our classification by *Forward Privacy* and *Backward Privacy*.

- *Forward Privacy (forward secure)*: Ensures that newly added data remains hidden to the server until it gets revealed by a later query, even if the server might have learned some secrets during previous queries.
- *Backward Privacy (backward secure)*: Search queries should not leak matching entries after they have been deleted.

Just 4 schemes of the 26 explicitly considered dynamic ones in the evaluated period of time are considered forward secure (Chang and Mitzenmacher, 2005; Stefanov et al., 2013; Bost, 2016; Bost et al., 2016) and up to 2016 there are none providing backward security (Cui et al., 2017).

3 DEVELOPMENT TRENDS IN SEARCHABLE ENCRYPTION

Finally, we connected our results to a timeline to find out if one can identify certain trends in the development of SE schemes. The outcome of these time-based results are described in this section.

We found out that the number of works on S/S and S/M schemes is increasing during the last 5 years of our observed set. On the other hand, the all-over trend for M/S is decreasing; M/M SE remains more or less on the same level.

Regarding cryptographic primitives we found a remarkable gap between asymmetric and symmetric settings from 2005 to 2010. After 2010 both keep on the same level with a slightly increasing trend. The first works on hybrid schemes were published in 2011 and the number of research work is slightly increasing - Figure 1 illustrates the trends of cryptographic primitives in SE schemes.

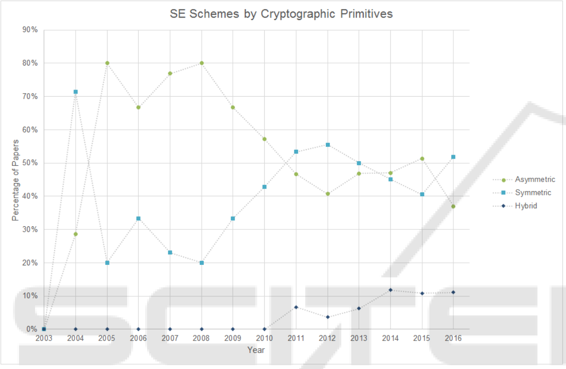

Figure 1: SE based on cryptographic primitives 2003-2016.

The SE search setting itself is significantly dominated by index-based settings. Full-domain search keeps constantly in low numbers. Figure 2 shows further increase in the future on SE schemes with an index-based search. The focus of the research is clearly on the efficiency of search queries, deducible from the increasing trend of using inverted indexes. Whereas the number of schemes utilizing forward index settings are slightly decreasing.

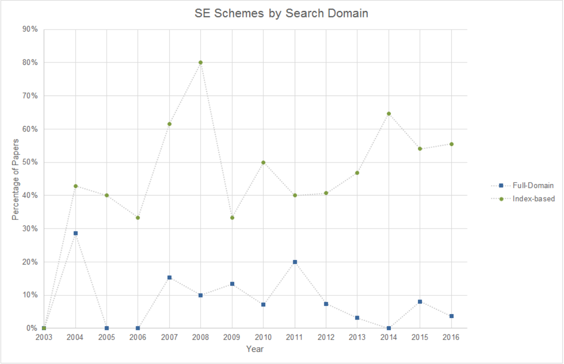

Figure 2: Full-domain vs. index-based SE settings 2003-2016.

We identified various search techniques, with the most popular being:

- boolean search,
- range search,
- ranked search, and
- fuzzy search.

The earliest scheme supporting boolean search queries appeared in 2003, this has been the most frequently engineered technique until 2014. Then ranked search became the most popular query expressiveness, which provides search results in a certain order. Ranked and fuzzy keyword search initially appeared in our paper collection in 2008/2009. Although fuzzy keyword search seems to be of significant importance for user's comfort, it only appears sporadically in articles about SE schemes, mainly due to the lack of efficient implementations. See also Figure 3 for a graphical depictions of the trends over time.

We also researched **cryptographic technologies** and identified identity based encrpytion (IBE), attribute based encryption (ABE), and bilinear maps (BM) as the dominating asymmetric techniques. However, from the corpus we could not determine a specific future trend for any of these techniques.

4 CONCLUSION AND FURTHER WORK

We initially started this project in order to find a specific searchable encryption scheme for a particular use case, but quickly became aware of the vast number of existing schemes, together with the lack of any current taxonomy of these schemes.

But usually one cannot decide on a particular scheme without being aware of the number of users and their read/write access privilege, resulting cryptographic primitives *and* the support of efficient and secure updates.

All the categorization domains we described in this paper are of significant importance for choosing the SE scheme for any use case.

We are currently in the process of implementing a framework that allows for fast and usable experimentation with different searchable encryption schemes in real-world scenarios (Haböck et al., 2018). In addition we try to finalize our categorisation of papers up until the current date, and try to identify future research trends from this.

The quite recently published file injection attacks on dynamic SE schemes and the definition of forward and backward security properties in the context

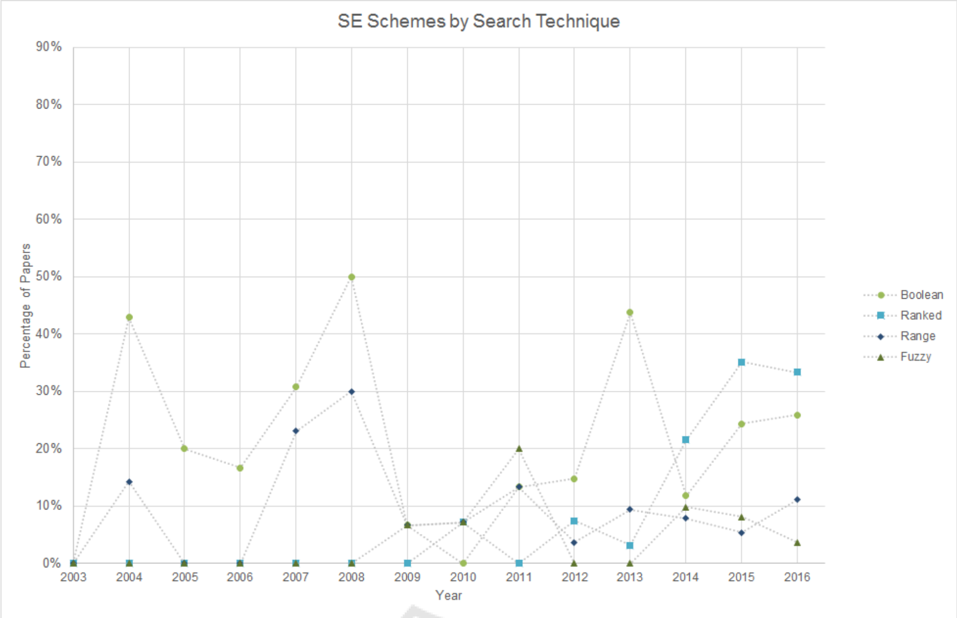

Figure 3: Supported Search Techniques 2003-2016.

of searchable encryption schemes evoke new challenges. be it by the need for thorough re-evaluation and re-design of already existing schemes or by designing new schemes with resilient security guarantees in real-world environments.

ACKNOWLEDGEMENTS

Part of our work on the Searchitect project is funded by *Netidee*[1] (*Internet Privatstiftung Austria*), grant no. 2099.

REFERENCES

Boneh, D., Crescenzo, G. D., Ostrovsky, R., and Persiano, G. (2004). Public Key Encryption with Keyword Search. In *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 506–522. IACR.

Boneh, D., Sahai, A., and Waters, B. (2011). Functional Encryption: Definitions and Challenges. In Ishai, Y., editor, *Theory of Cryptography: 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, pages 253–273. Springer Berlin Heidelberg, Berlin, Heidelberg.

Bösch, C., Hartel, P., Jonker, W., and Peter, A. (2014). A Survey of Provably Secure Searchable Encryption. *ACM Computing Surveys*, 47(2):1–51.

Bost, R. (2016). Sophos - Forward Secure Searchable Encryption. Published: Cryptology ePrint Archive, Report 2016/728.

Bost, R., Fouque, P.-A., and Pointcheval, D. (2016). Verifiable Dynamic Symmetric Searchable Encryption: Optimality and Forward Security. Published: Cryptology ePrint Archive, Report 2016/062.

Cash, D., Grubbs, P., Perry, J., and Ristenpart, T. (2015). Leakage-Abuse Attacks Against Searchable Encryption. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 668–679, Denver, Colorado, USA. ACM.

Chang, Y.-c. and Mitzenmacher, M. (2005). Privacy Preserving Keyword Searches on Remote Encrypted Data. In *ACNS 2005*, volume 3531 of *LNCS*, pages 442–455. Springer Berlin Heidelberg.

Cui, S., Asghar, M., Galbraith, S., and Russello, G. (2017). Secure and Practical Searchable Encryption: A Position Paper. In Pieprzyk, J. and Suriadi, S., editors, *Information Security and Privacy: 22nd Australasian Conference, ACISP 2017, Auckland, New Zealand, July 5, 2017, Proceedings, Part I*, pages 266–281. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-60055-0_14.

Haböck, U., Koschuch, M., Kramer, I., Schmidt, S., and Tausig, M. (2018). Searchitect - A developer framework for hybrid searchable encryption (position paper). In *Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security, IoTBDS 2018, Funchal, Madeira, Portugal, March 19-21, 2018.*, pages 291–298.

Islam, M. S., Kuzu, M., and Kantarcioglu, M. (2012).

[1] https://www.netidee.at

Access Pattern disclosure on Searchable Encryption: Ramification, Attack and Mitigation. In *NDSS*.

Liesdonk, P. V., Sedghi, S., Doumen, J., Hartel, P., and Jonker, W. (2010). Computationally Efficient Searchable Symmetric Encryption. In *SIAM Conference on Data Mining SDM 2010*, volume 6358 of *LNCS*, pages 87–100. Springer.

Liu, C., Zhu, L., Wang, M., and Tan, Y.-a. (2013). Search Pattern Leakage in Searchable Encryption: Attacks and New Constructions. *IACR Cryptology ePrint Archive*.

M. Li, S. Yu, K. Ren, W. Lou, and Y. T. Hou (2013). Toward Privacy-Assured and Searchable Cloud Data Storage Services. *IEEE Network*, 27(4):56–62.

Naveed, M., Kamara, S., and Wright, C. V. (2015). Inference Attacks on Property-Preserving Encrypted Databases. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 644–655, Denver, Colorado, USA. ACM.

Sedghi, S., Doumen, J., Hartel, P., and Jonker, W. (2008). Towards an Information Theoretic Analysis of Searchable Encryption. In Chen, L., Ryan, M. D., and Wang, G., editors, *Information and Communications Security: 10th International Conference, ICICS 2008 Birmingham, UK, October 20 - 22, 2008 Proceedings*, pages 345–360. Springer Berlin Heidelberg, Berlin, Heidelberg.

Song, D. X., Wagner, D., and Perrig, A. (2000). Practical techniques for searches on encrypted data. In *S&P 2000*, pages 44–55. IEEE.

Stefanov, E., Papamanthou, C., Shi, E., and Encryption, S. (2013). Practical Dynamic Searchable Encryption with Small Leakage.

Tang, Q. (2012). Search in Encrypted Data: Theoretical Models and Practical Applications. *IACR Cryptology ePrint Archive*, 2012(648).

Wang, C., Cao, N., Li, J., Ren, K., and Lou, W. (2010). Secure Ranked Keyword Search over Encrypted Cloud Data. *2010 IEEE 30th International Conference on Distributed Computing Systems*, pages 253–262.

Zhang, Y., Katz, J., and Papamanthou, C. (2016). All Your Queries Are Belong to Us: The Power of File-Injection Attacks on Searchable Encryption. *IACR Cryptology ePrint Archive*, 2016:172.