Literature Review

Mireya Jurado

December 22, 2017

Contents

Author	Title	Date
Fuller et al.	SoK: Cryptographically Protected Database Search	10/31/17
Popa et al.	CryptDB: Protecting Confidentiality with Encrypted Query Processing	12/04/17
Naveed, Kamara, and Wright	Inference attacks on property-preserving encrypted databases	12/07/17
Popa, Zeldovich, and Balakrishnan	Guidelines for Using the CryptDB System Securely	12/08/17
Boldyreva, Chenette, and O'Neill	Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions.	TODO
Boldyreva and Chenette	Efficient fuzzy search on encrypted data	TODO
Song, Wagner, and Perrig	Practical techniques for searches on encrypted data	TODO
Akin and Sunar	On the difficulty of securing web applications using CryptDB	TODO
Pouliot and Wright	The shadow nemesis: Inference attacks on efficiently deployable, efficiently searchable encryption	TODO
Cash et al.	Dynamic Searchable Encryption in Very-Large Databases: Data Structures and Implementation.	TODO
Cash et al.	Leakage-abuse attacks against searchable encryption	TODO
Chenette et al.	Practical order-revealing encryption with limited leakage	TODO
Pappas et al.	Blind seer: A scalable private dbms	TODO
Alvim et al.	Additive and multiplicative notions of leakage, and their capacities	TODO
Alvim et al.	Axioms for information leakage	TODO
Alvim et al.	Measuring information leakage using generalized gain functions	TODO
McIver et al.	Abstract channels and their robust information-leakage ordering	TODO
Boneh et al.	Public key encryption with keyword search	TODO

Table 1: The papers organized by date finished. Note: These are not ordered yet

Roles		Operations	
Provider	provides and modifies data	Init	server obtains database from provider
Server	handles storage and processing	Query	querier provides query to server if allowed
Querier	wishes to learn things about the data	Update	provider gives set up updates to server
Enforcer	ensures that rules are applied	Refresh	server obtains new DB from provider
Authorizer	specifies data- and query-based rules		with same data but better security
Base Queries	Some Techniques	Objects vulnerable to leakage	
Equality	DET, Inverted Index, Tree Traversal	Data items or indexing data structure	
Boolean	Inverted Index	Queries	
Range	OPE	Records returned in response to queries	
		Access control rules	

Table 2: Database Basics

1 SoK Cryptographically Protected Database Search

Benjamin Fuller et al. "SoK: Cryptographically Protected Database Search". In: (2017) 2017 IEEE Security and Privacy

1.1 Summary

Fuller et al. provide a summary of current knowledge on protected database search systems. They outline primitive operations, current implementations, an analysis of attacks, and tools for creating a protected database. This paper serves as a reference on protected search databases, security, performance, and usability of base queries, and current leakage inference attacks.

1.2 Authors

This Systemization of Knowledge paper was written recently by MIT Lincoln Laboratory's Group 53. Ben Fuller spearheaded this effort from the University of Connecticut and Mayank is at Boston University. Work was funded by an Air Force Contract.

1.3 Outline

- Argument: There are trade-offs between the different approaches to protected database search systems
- Protective Mechanisms:
  - Legacy: Order Preserving Encryption (OPE) and Deterministic Encryption (DET). Systems encrypt equality data with DET and range with OPE
  - Inverted Index: use a reverse look up table that maps keywords to file IDs [12] [10]
  - Tree Traversal: hide access pattern in tree data structure [15] [24]
  - Oblivious RAM: obfuscate memory access pattern [14] [25] [22]
- Further Reading:
  - Includes a list of open problems
  - Lists 11 different database attacks
- Definitions/Key Words:
  - See Table 2 for database roles, operations, base queries, and objects vulnerable to leakage

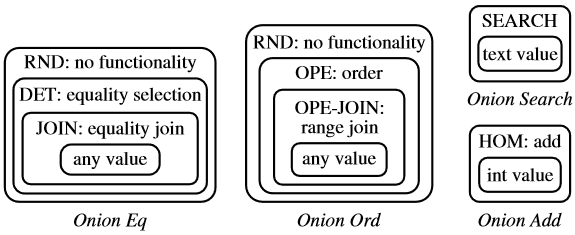

Figure 1: The four onions. Figure directly from Popa et al.

2 CryptDB

Raluca Ada Popa et al. "CryptDB: Protecting Confidentiality with Encrypted Query Processing". In: *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*. ACM. 2011, pp. 85–100

2.1 Summary

Popa et al. introduce their encrypted database, CryptDB. This database is able to execute SQL queries over encrypted data through three key ideas. Firstly, each data item is encrypted in a way that allows relevant SQL queries to be performed. Secondly, CryptDB uses *onions of encryption* to dynamically adjust the query-based encryption (See Figure 1). Lastly, encryption keys are chained to user passwords to allow granular access control.

2.2 Authors

Popa did this work during her PhD at MIT under Zeldovich, with Balakrishnan (systems). She's now at UC Berkley. Work was supported by an NSF grant and by Google.

2.3 Outline

- Argument: CryptDB can protect data by executing SQL queries over encrypted data
- Findings
  - Motivation: protect online applications
  - Threat model: Does not ensure integrity, freshness, or completeness of results
    1. Curious database administrator: full access but passive
    2. Adversary that gains control of application and DBMS servers (worst case).
  - Challenges
    1. Minimize amount of confidential info to DBMS server but execute a variety of queries
    2. Minimize data leaked when adversary compromises the application server and the DBMS server
  - Solution
    1. SQL-aware encryption strategy: all SQL queries made up of a defined set of primitive operators. Symmetric-key, unmodified DBMS
    2. Adjustable query-based encryption: adjusts encryption scheme per given data item depending on queries observed at run time $\rightarrow$ Onions of encryption
    3. Chain encryption keys to user passwords
- How it works: Figure 2
  - We trust the application and the proxy (Threat 1)
  - Proxy stores a secret master key MK, the database schema, and the current encryption layer of all columns.
  - Database sees schema (anatomized), encrypted user data, and some auxiliary tables

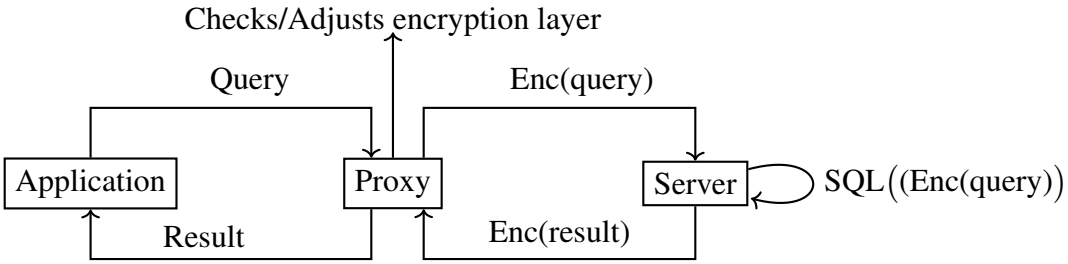

Figure 2: CryptDB Steps

Encryption	Description	Use
Random (RND)	Traditional encryption	No computation performed efficiently
Deterministic (DET)	If $x = y$, $C(x) = C(y)$	Equality checks
Order-reserving Encryption (OPE)	If $x < y$, $C(x) < C(y)$	Range queries
Homomorphic encryption (HOM)	Pallier multiplication	Addition
Join and OPE-Join	Keyed cryptographic hash	Column joins
Search	Token of keyword	Keyword searching

Table 3: 6 cryptographic tools

- Definitions
  - User defined functions (UDF): enable the server to compute on cipher texts for certain operations
- Further Reading: Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions.
- Critique
  - In their presentation, they describe the data structure as order preserving *encoding*, which might invalidate security guarantees
  - They do not include a formal security proof
  - The layer of security is never added back on after it has been removed. After a certain amount of time, the database will be encrypted with the lowest level of security, which allows for frequency attacks (See Inference attacks on property-preserving encrypted databases).

Column	Attack	Description
DTE	Frequency Analysis	Correlate histograms of encrypted and axillary databases
	l_p-optimization	Correlate rank and *amplitude* in respective histograms
OPE	Sorting Attack	Sorts dense columns, maps them to element of message space with same rank
	Cumulative Attack	Sort ciphertext, then correlate with histogram frequencies

Table 4: Four Inference Attacks

3 Inference Attacks

Muhammad Naveed, Seny Kamara, and Charles V Wright. "Inference attacks on property-preserving encrypted databases". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2015, pp. 644–655

3.1 Summary

Naveed et al. describe four attacks that successfully recovered real patient data from 200 U.S. hospitals from CryptDB. In CryptDB, after an equality query is given to a compatible column, that column is encrypted with deterministic encryption (DTE) and leaks equality. Similarly, a range query results in a column encrypted with order-preserving encryption (OPE) that leaks order. Naveed et al. demonstrate that this leakage can be attacked by using publicly available axillary databases (see Table 4).

3.2 Authors

Naveed did this work at the University of Illinois Urbana-Champaign. He's now at University of Southern California. Kamara was a researcher at Microsoft and is now a professor at Brown. Wright is an assistant professor at Portland State and was technical staff at MIT Lincoln Laboratory from 2008 to 2012.

3.3 Outline

- Argument: Encrypted databases should not be used for electronic medical records
- Threat Model: adversary has access to the encrypted database but can't influence it or see queries.
  - Individual attacks: adversary wants information on a row
  - Aggregate attacks: adversary wants statistical info on the whole database
- Results
  - OPE attributes (such as age and disease severity) recovered for more than 80% of patient records from 95% of the hospitals
  - DE attributes (such as sex, race, and mortality risk) recovered for more than 60% of the records from more than 60% of the hospitals.
  - Note: they do not attack the weakest encryption schemes of CryptDB
- Definitions
  - Frequency analysis: involved creating a histogram of the encrypted database and comparing it to the histogram of the axillary database
  - Inference attacks: combine leakage with publicly available information
- Further Reading:"On the difficulty of securing web applications using CryptDB"
- Critique: Frequency analysis uses a publicly accessible auxiliary database, but Naveed et al. use a private database that has the exact same columns and possible values as their target database. A real adversary would not likely have access to an auxiliary database of this quality.

4 Guidelines for Using CryptDB

Raluca Ada Popa, Nickolai Zeldovich, and Hari Balakrishnan. "Guidelines for Using the CryptDB System Securely". In: *IACR Cryptology ePrint Archive* 2015 (2015), p. 979

4.1 Summary

Popa et al. rebut the attacks done by [17] by claiming that they implemented the system incorrectly. They state that the study performed by Naveed et al. "represents an unsafe usage of CryptDB" and that had the guidelines been appropriately followed, none of the attacks would have been successful. Specifically, they argue that Naveed et al. used DET encryption on columns with repeating fields, violating CryptDB usage guidelines.

4.2 Authors

This rebuttal was made by the same authors of the original CryptDB paper with the exception of Redfield.

4.3 Outline

- Argument: If CryptDB is used correctly, it is still safe
  - Sensitive columns should be marked sensitive. This ensures that for equality queries,
    - A column with unique fields uses DET. If fields are UNIQUE. $\rightarrow$ flat histogram $\rightarrow$ no frequency analysis attacks
    - A column with repeating fields uses SEARCH
  - Many queries can still work with the strongest RND encryption, including SELECT
- Further Reading:
  - Kamara: Attacking encrypted database systems, blog post, Sept 7, 2015
  - Her dissertation: Building practical systems that compute on encrypted data
- Critique
  - The SEARCH technique that is used for checking equality on repeating fields is based off of key word tokens. How do the tokens not leak equality?
  - They argue that columns encrypted with RND can still perform SELECT queries. However, SELECT [...] WHERE [...] often requires knowledge concerning equality or range

References

[1] Ihsan H Akin and Berk Sunar. "On the difficulty of securing web applications using CryptDB". In: *Big Data and Cloud Computing (BdCloud), 2014 IEEE Fourth International Conference on*. IEEE. 2014, pp. 745–752.

[2] Mário S Alvim et al. "Additive and multiplicative notions of leakage, and their capacities". In: *Computer Security Foundations Symposium (CSF), 2014 IEEE 27th*. IEEE. 2014, pp. 308–322.

[3] Mário S Alvim et al. "Axioms for information leakage". In: *Computer Security Foundations Symposium (CSF), 2016 IEEE 29th*. IEEE. 2016, pp. 77–92.

[4] Mário S Alvim et al. "Measuring information leakage using generalized gain functions". In: *Computer Security Foundations Symposium (CSF), 2012 IEEE 25th*. IEEE. 2012, pp. 265–279.

[5] Alexandra Boldyreva and Nathan Chenette. "Efficient fuzzy search on encrypted data". In: *International Workshop on Fast Software Encryption*. Springer. 2014, pp. 613–633.

[6] Alexandra Boldyreva, Nathan Chenette, and Adam O'Neill. "Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions." In: *CRYPTO*. Vol. 6841. Springer. 2011, pp. 578–595.

[7] Dan Boneh et al. "Public key encryption with keyword search". In: *Eurocrypt*. Vol. 3027. Springer. 2004, pp. 506–522.

[8] David Cash et al. "Dynamic Searchable Encryption in Very-Large Databases: Data Structures and Implementation." In: *NDSS*. Vol. 14. 2014, pp. 23–26.

[9] David Cash et al. "Leakage-abuse attacks against searchable encryption". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2015, pp. 668–679.

[10] Melissa Chase and Seny Kamara. "Structured encryption and controlled disclosure". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2010, pp. 577–594.

[11] Nathan Chenette et al. "Practical order-revealing encryption with limited leakage". In: *International Conference on Fast Software Encryption*. Springer. 2016, pp. 474–493.

[12] Reza Curtmola et al. "Searchable symmetric encryption: improved definitions and efficient constructions". In: *Journal of Computer Security* 19.5 (2011), pp. 895–934.

[13] Benjamin Fuller et al. "SoK: Cryptographically Protected Database Search". In: (2017).

[14] Oded Goldreich and Rafail Ostrovsky. "Software protection and simulation on oblivious RAMs". In: *Journal of the ACM (JACM)* 43.3 (1996), pp. 431–473.

[15] Seny Kamara and Charalampos Papamanthou. "Parallel and dynamic searchable symmetric encryption". In: *International Conference on Financial Cryptography and Data Security*. Springer. 2013, pp. 258–274.

[16] Annabelle McIver et al. "Abstract channels and their robust information-leakage ordering". In: *International Conference on Principles of Security and Trust*. Springer. 2014, pp. 83–102.

[17] Muhammad Naveed, Seny Kamara, and Charles V Wright. "Inference attacks on property-preserving encrypted databases". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2015, pp. 644–655.

[18] Vasilis Pappas et al. "Blind seer: A scalable private dbms". In: *Security and Privacy (SP), 2014 IEEE Symposium on*. IEEE. 2014, pp. 359–374.

[19] Raluca Ada Popa, Nickolai Zeldovich, and Hari Balakrishnan. "Guidelines for Using the CryptDB System Securely". In: *IACR Cryptology ePrint Archive* 2015 (2015), p. 979.

[20] Raluca Ada Popa et al. “CryptDB: Protecting Confidentiality with Encrypted Query Processing”. In: *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*. ACM. 2011, pp. 85–100.

[21] David Pouliot and Charles V Wright. “The shadow nemesis: Inference attacks on efficiently deployable, efficiently searchable encryption”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2016, pp. 1341–1352.

[22] Daniel S Roche, Adam Aviv, and Seung Geol Choi. “A practical oblivious map data structure with secure deletion and history independence”. In: *Security and Privacy (SP), 2016 IEEE Symposium on*. IEEE. 2016, pp. 178–197.

[23] Dawn Xiaoding Song, David Wagner, and Adrian Perrig. “Practical techniques for searches on encrypted data”. In: *Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on*. IEEE. 2000, pp. 44–55.

[24] Emil Stefanov, Charalampos Papamanthou, and Elaine Shi. “Practical Dynamic Searchable Encryption with Small Leakage.” In: *NDSS*. Vol. 14. 2014, pp. 23–26.

[25] Emil Stefanov et al. “Path ORAM: an extremely simple oblivious RAM protocol”. In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM. 2013, pp. 299–310.