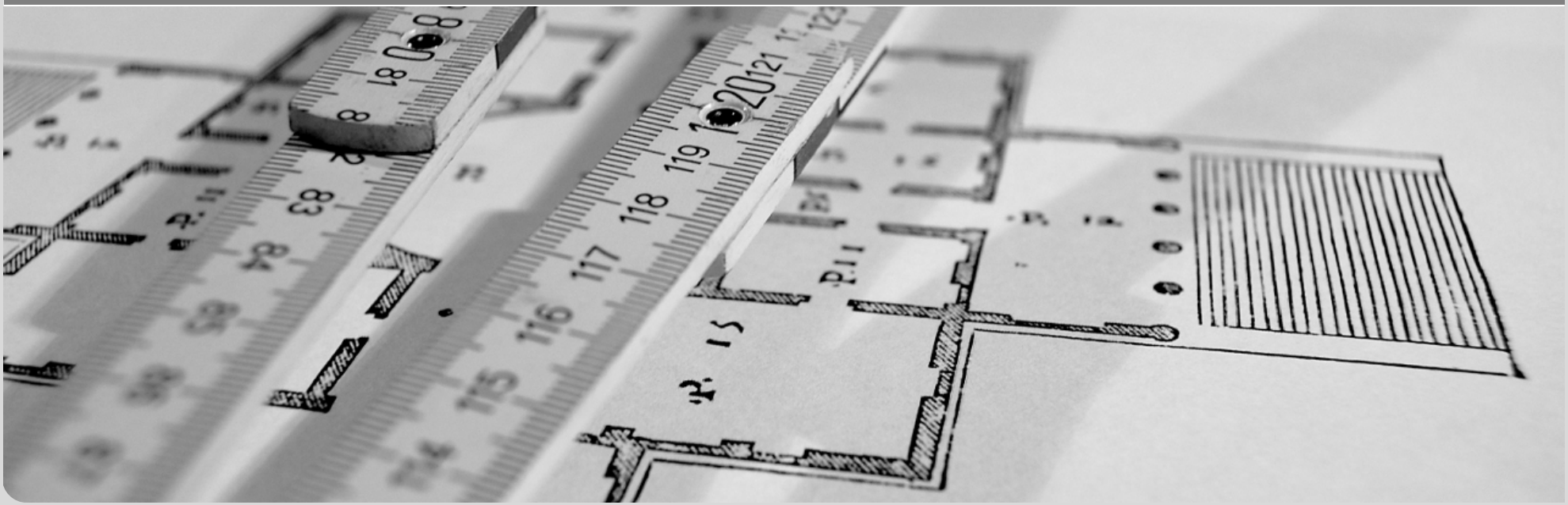
Karlsruher Institut für Technologie

Closing the Gap: A Universal Privacy Framework for Outsourced Data

Dirk Achenbach, Matthias Huber, Jörn Müller-Quade, Jochen Rill | 3. September 2015

INSTITUT FÜR KRYPTOGRAPHIE UND SICHERHEIT



Context of this Work

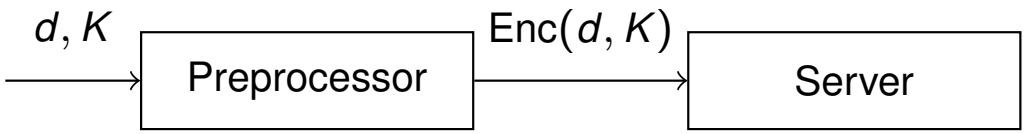
Imagine: Facebook wants to host its user database on Amazon's cloud service platform.

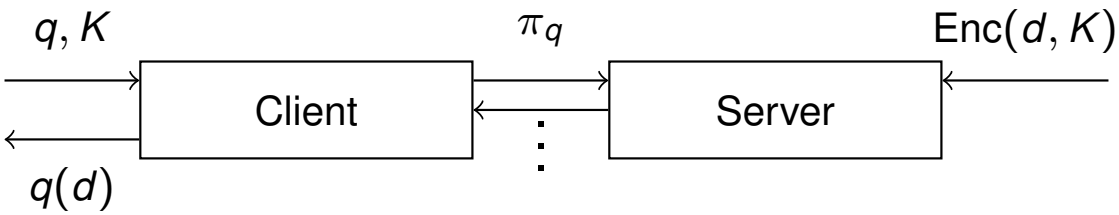
- The user database is huge and contains highly sensitive information, maybe even company secrets.
- Queries made to the outsourced database must be very efficient (therefore simple encryption is not an option).

Context of this Work

We need to investigate the security (using formal models) of efficient solutions for data outsourcing.

A Model for Outsourced Data





Definitions

$\mathfrak{S}$	Server
$\mathfrak{C}$	Client
$\mathcal{Q}$	Set of all allowed queries
$\pi \in \mathcal{Q}$	One query from the set
$\text{view}_{\mathfrak{S}}^{\pi_j}(\text{Enc}(d, K))$	Server view for query π_j on encrypted database d
$(\text{Gen}, \text{Enc}, \text{Dec}, \mathcal{Q})$	A queryable outsourcing scheme

Existing Solutions and their Security: PIR

Karlsruher Institut für Technologie

Definition (Private Information Retrieval)

A queryable outsourcing scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \{\pi\})$ exhibits *Computational Single-Server Private Information Retrieval* (PIR) when two conditions hold for any $n \in \mathbb{N}$, any security parameter $k \in \mathbb{N}$, and any data set d over $\Sigma = \{0,1\}^n$:

1. Correctness: $\forall i \in \{0, \ldots, n-1\} : \pi_i^{\mathfrak{C}}(d) = d[i]$.
2. Privacy: $\forall c \in \mathbb{N}, i, j \in \{0, \ldots, n-1\}, \forall \mathcal{A} \exists K \in \mathbb{N}$ such that $\forall k > K$

$$|\Pr[\mathcal{A}(\mathsf{view}_{\mathfrak{C}}^{\pi_i}(\mathsf{Enc}(d, K))) = 1] - \Pr[\mathcal{A}(\mathsf{view}_{\mathfrak{C}}^{\pi_j}(\mathsf{Enc}(d, K))) = 1]| < \frac{1}{\max(k, n)^c}.$$

Existing Solutions and their Security: PIR

PIR is great, since it guarantees that queries are completely hidden . . .

Existing Solutions and their Security: PIR

PIR is great, since it guarantees that queries are completely hidden . . .

. . . but PIR can not be realised efficiently . . .

. . . and gives no guarantees concerning the data.

Existing Solutions and their Security: Goh's construction [3]

Security Game ($\text{IND-CKA}^{\mathcal{A}}_{(\text{Gen},\text{Enc},Q)}$)

1. *The experiment chooses a key* $K \leftarrow \text{Gen}(1^k)$ *and a random bit* $b \leftarrow \{0,1\}$.
2. *The adversary* $\mathcal{A}$ *is given input* 1^k *and access to an oracle* $\text{view}^{\pi}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$.
3. $\mathcal{A}$ *outputs two plaintexts* m_0 *and* m_1 *of the same length to the experiment. The adversary must not have queried the oracle for words that are only in one of the two plaintexts.*
4. $\mathcal{A}$ *is given* $\text{Enc}_K(m_b)$ *and access to an oracle* $\text{view}^{\pi}_{\mathfrak{S}}(\text{Enc}(m_b, K))$
5. $\mathcal{A}$ *submits a guess* b' *for* b.

The result of the experiment is 1 *if* $b' = b$ *and* 0 *else.*

Existing Solutions and their Security: Goh's construction [3]

IND-CKA restricts the adversary to allow for an efficient realisation . . .

Existing Solutions and their Security: Goh's construction [3]

IND-CKA restricts the adversary to allow for an efficient realisation . . .

. . . but in a very specific way which may no be suited for other schemes . . .

. . . and it also gives no guarantees concerning queries.

Existing Solutions and their Security

There are multiple different and specifically tailored security notions for outsourcing schemes. They are hiding parts of the data, parts of the queries, or a combination of both.

Existing Solutions and their Security

This makes comparing different schemes nearly impossible!

. . . so we need a general framework to instantiate security notions from, which fits more schemes.

A Unified Framework

What do we even want to hide?

- Queries: Query Privacy
- Results: Result Privacy
- The data itself: Data Privacy

A Unified Framework: Privacy of Queries

Security Game ($\text{Q-IND}^{\mathcal{A}}_{(\text{Gen},\text{Enc},\text{Q})}(k)$)

1. *The experiment chooses a key* $K \leftarrow \text{Gen}(1^k)$.
2. $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$.
3. $(q_0, q_1) \leftarrow \mathcal{A}$. *$q_0$ and q_1 must yield protocols π_{q_0} and π_{q_1} with the same number of protocol messages.*
4. *The experiment draws a random bit* $b \leftarrow \{0, 1\}$.
5. *Challenge oracle:* $\mathcal{A} \leftarrow \text{view}^{\pi_{q_b}}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *The oracle takes any data set $d \in \Delta$ as input, internally runs the protocol π_{q_b} on* $\text{Enc}(d, K)$, *and outputs* $\text{view}^{\pi_{q_b}}_{\mathfrak{S}}(\text{Enc}(d, K))$ *to the adversary.*
6. $\mathcal{A}$ *outputs a guess b' for b.*

The result of the experiment is 1 *if* $b' = b$ *and* 0 *else.*

A Unified Framework: Privacy of Queries

This is equivalent to PIR!

A Unified Framework: Privacy of Queries

Security Game ($\text{Q-IND}^{\mathcal{A},R_q,n_1,n_2,n_3}_{(\text{Gen},\text{Enc},\text{Q})}(k)$)

1. *The experiment chooses a key* $K \leftarrow \text{Gen}(1^k)$.
2. $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *$\mathcal{A}$ is only allowed to query the oracle for a total number of* n_1 *times.*
3. $(q_0, q_1) \leftarrow \mathcal{A}$. *$(q_0, q_1)$ is restricted with regard to equivalence relation* $R_q \subseteq \Pi^2$, *i.e.* $(q_0, q_1) \in R_q$.
4. *The experiment draws a random bit* $b \leftarrow \{0, 1\}$.
5. *Challenge oracle:* $\mathcal{A} \leftarrow \text{view}^{\pi_{q_b}}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *$\mathcal{A}$ may call the challenge oracle for a total number of* n_2 *times.*
6. *Run oracle:* $\mathcal{A} \leftarrow \text{run}^{\pi_b}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *The run oracle executes queries just as the view oracle does, but has no output. $\mathcal{A}$ is allowed to call the run oracle for a total number of* n_3 *times.*
7. $\mathcal{A}$ *outputs a guess* b' *for* b.

A Unified Framework: Privacy of Data

Security Game ($\text{D-IND}^{\mathcal{A}}_{(\text{Gen},\text{Enc},\text{Q})}(k)$)

1. *The experiment chooses a key* $K \leftarrow \text{Gen}(1^k)$.
2. $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *The oracle takes a query* q *and a data set* d *as input and returns* $\text{view}^{\pi_q}_{\mathfrak{S}}(\text{Enc}(d, K))$.
3. $(d_0, d_1) \leftarrow \mathcal{A}$ *(both of equal length).*
4. *The experiment draws a random bit* $b \leftarrow \{0, 1\}$.
5. *Challenge oracle:* $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(d_b, K))$. *That is, the oracle takes any query* q *such that* $\pi_q \in \mathcal{Q}$ *as input, internally runs the protocol* π_q *on* $\text{Enc}(d_b, K)$*, and outputs* $\text{view}^{\pi_q}_{\mathfrak{S}}(\text{Enc}(d_b, K))$ *to the adversary.*
6. $\mathcal{A}$ *outputs a guess* b' *for* b.

The result of the experiment is 1 *if* $b' = b$ *and* 0 *else.*

A Unified Framework: Privacy of Data

Security Game $(\text{D-IND}^{\mathcal{A},R_d,n_1,n_2,n_3}_{(\text{Gen},\text{Enc},\text{Q})}(k))$

1. *The experiment chooses a key* $K \leftarrow \text{Gen}(1^k)$.
2. $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(\cdot, K))$. *$\mathcal{A}$ is only allowed to query the oracle for a total number of* n_1 *times.*
3. $(d_0,d_1) \leftarrow \mathcal{A}$. *$(d_0,d_1)$ is restricted with regard to equivalence relation* $R_d \subseteq \Delta^2$, *i.e.* $(d_0, d_1) \in R_d$
4. *The experiment draws a random bit* $b \leftarrow \{0,1\}$.
5. *Challenge oracle:* $\mathcal{A} \leftarrow \text{view}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(d_b, K))$. *$\mathcal{A}$ may call the challenge oracle for a total number of* n_2 *times.*
6. *Run oracle:* $\mathcal{A} \leftarrow \text{run}^{\pi_\cdot}_{\mathfrak{S}}(\text{Enc}(d_b, K))$. *The run oracle executes queries just as the view oracle does, but has no output. $\mathcal{A}$ is allowed to call the run oracle for a total number of* n_3 *times.*
7. *$\mathcal{A}$ outputs a guess* b' *for* b.

A Unified Framework

- Q-IND $\not\Longleftrightarrow$ D-IND
- D-IND $\wedge$ Q-IND $\Longleftrightarrow$ R-IND

A Unified Framework

- PIR (Chor et. al [1]) $\iff$ Q-IND
- Ind_{SSE} (Curtmola et al. [2]) $\implies$ D-IND
- D-IND $\wedge$ Q-IND $\implies$ Ind_{SSE}
- D-IND $\implies$ IND-CKA
- $\text{PrivK}^{\text{cppa}}_{\mathcal{A},\text{Enc}}(k)$ (Haynberg et al. [4]) $\iff$ D-IND
- IND-ICP (Huber et al. [5]) $\iff$ Static Security

Conclusion

- We strictly separated and formalised security goals for data outsourcing.
- We generalised our formalisation to capture mechanisms with weaker security properties.
- Our framework is applicable for all outsourcing schemes (i.e. searchable encryption or database outsourcing).
- Our framework makes the security of different schemes compareable.

Future Work

- Investigate simulation based formulations and their relation to game based notions.
- Investigate relations among the generalised notions.
- Investigate limitations of “pure” Data Privacy for practical outsourcing schemes.

Bibliography I

Benny Chor u. a. “Private Information Retrieval”. In: *J. ACM* 45.6 (Nov. 1998), S. 965–981. ISSN: 0004-5411. DOI: 10.1145/293347.293350. URL: http://doi.acm.org/10.1145/293347.293350.

Reza Curtmola u. a. “Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions”. In: *Proceedings of the 13th ACM Conference on Computer and Communications Security*. CCS ’06. Full version available at https://eprint.iacr.org/2006/210. Alexandria, Virginia, USA: ACM, 2006, S. 79–88. ISBN: 1-59593-518-5. DOI: 10.1145/1180405.1180417.

Bibliography II


Eu-Jin Goh u. a. “Secure Indexes.” In: *IACR Cryptology ePrint Archive* 2003 (2003). https://eprint.iacr.org/2003/216/, S. 216.


Rolf Haynberg u. a. “Symmetric Searchable Encryption for Exact Pattern Matching using Directed Acyclic Word Graphs.” In: *SECRYPT*. 2013, S. 403–410.


Matthias Huber u. a. “Cumulus4j: A Provably Secure Database Abstraction Layer”. In: *CD-ARES Workshops*. 2013, S. 180–193.