THE MAGIC WORLD OF SEARCHABLE ENCRYPTION

CHRISTIAN FORLER

TOBIAS MUELLER
General Scenario
General Scenario

User encrypts data, sends it to a server, forgets about it, then wants to search it for, e.g. substrings
General Scenario
Why...? - Ideas?
Approaches
Can we do better?
Index based
Outlook
Conclusions
Why...? - Ideas?
Motivation

- Emails
Motivation

- Emails
- Documents
Example: Contacts
Example: More Concrete

Store

<table>
<thead>
<tr>
<th>First Name</th>
<th>Last Name</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>Foo</td>
<td>123</td>
</tr>
<tr>
<td>Bob</td>
<td>Foo</td>
<td>345</td>
</tr>
<tr>
<td>Eve</td>
<td>Bar</td>
<td>456</td>
</tr>
</tbody>
</table>

*securely in the cloud™*
Now, the server knows your contacts. :-(

```plaintext
Client

Plaintext

W₁
W₂
.
.
Wₙ

Server

Plaintext

W₁
W₂
.
.
Wₙ

Wᵢ
```
GAME OVER
ENCRYPT

ALL THE THINGS
Encrypt all the things!

**Simple Crypto**

- Plaintext: $W_1, W_2, \ldots, W_n$
- Client: $SecEnc_k(W_i) \rightarrow C_i$
- Server: $C_1, C_2, \ldots, C_n$
REALLY DOWNLOAD

ALL THE THINGS?
Simple Crypto - Search

Client

\( \text{Decrypt}_K(C_i) \)

\( W_i \) \( \equiv \) \( C_i \)

Server

\( C_1 \)
\( C_2 \)
\( \ldots \)
\( C_n \)
Can we do better?
Deterministic Encryption of Keywords - Setup

Plaintext

\[ W_1, W_2, \ldots, W_n \]

\[ W_i \]

\[ \text{DetEnc}_k(W_i) \]

\[ C_i \]

Ciphertext

\[ C_1, C_2, \ldots, C_n \]
Deterministic Encryption of Keywords - Search

\[ \text{DetEnc}_k(W_i) \]

\[ W_i \rightarrow C_i \]

\[ C_i = C_i \]

\[ C_1, C_2, \ldots, C_n \]
Deterministic Encryption of Keywords - Problem
Deterministic Encryption of Keywords - Problem

Deterministic encryption sucks!
Keyword based - Setup (Song, Wagner, Perrig)

Encrypt-then-Mask

Search key $k_i = H_k(L_i)$

Magic Mask: $T_i$ can be derived from $S_i$, i.e. $T_i = H_{k_i}(S_i)$
Keyword based - Search

Client

\[ \text{Encrypt} \]

\[ W_i \rightarrow \text{DetEnc}_k(W_i) \]

\[ L_i \quad R_i \]

Server

\[ \text{Ciphertext} \]

\[ C_1 \quad C_2 \quad \ldots \quad C_n \]

\[ C_i \]

\[ X_i \quad Y_i \]

\[ \text{Test: } H_{K_i}(X_i) == Y_i \]
Speed

Plaintext size (King James Bible): 4.3 MB

Ciphertext size: 25 MB

Time to encrypt: 0.211 sec

Search (in seconds):
- Foobar 0.181
- God 0.003
- towel: 0.155
- Eve 0.005
- wrath 0.014
- dragon 0.094
### Plaintext Index - Search

<table>
<thead>
<tr>
<th>Client</th>
<th>Index</th>
<th>Server</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Plaintext</strong></td>
<td><strong>Value</strong></td>
<td><strong>Ciphertext</strong></td>
</tr>
<tr>
<td>1  Alice</td>
<td>Foo</td>
<td>Enc(_k)(Alice, Foo)</td>
</tr>
<tr>
<td>2  Bob</td>
<td>Foo</td>
<td>Enc(_k)(Bob, Foo)</td>
</tr>
<tr>
<td>3  Eve</td>
<td>Bar</td>
<td>Enc(_k)(Eve, Bar)</td>
</tr>
</tbody>
</table>

#### Token Value Table

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(Foo)</td>
<td>[1, 2]</td>
</tr>
<tr>
<td>H(Bar)</td>
<td>[3]</td>
</tr>
</tbody>
</table>
Plaintext Index - Hell of Synchronisation

AND THEN I SAID

SURE YOU CAN Synchronise
Enc. Index based - Setup

Client

Plaintext

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(good)</td>
<td>[1, 2]</td>
</tr>
<tr>
<td>H(evil)</td>
<td>[3, 4]</td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(good)</td>
<td>[1, 2]</td>
</tr>
<tr>
<td>H(evil)</td>
<td>[3, 4]</td>
</tr>
</tbody>
</table>

Server

Ciphertext

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enc_k(Alice)</td>
<td></td>
</tr>
<tr>
<td>Enc_k(Bob)</td>
<td></td>
</tr>
<tr>
<td>Enc_k(Eve)</td>
<td></td>
</tr>
<tr>
<td>Enc_k(Mallory)</td>
<td></td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(good)</td>
<td>[1, 2]</td>
</tr>
<tr>
<td>H(evil)</td>
<td>[3, 4]</td>
</tr>
</tbody>
</table>
Enc. Index based - Search

Client

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(good)</td>
<td>[ 1, 2 ]</td>
</tr>
<tr>
<td>H(evil)</td>
<td>[ 3, 4 ]</td>
</tr>
</tbody>
</table>

Server

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H(good)</td>
<td>[ 1, 2 ]</td>
</tr>
<tr>
<td>H(evil)</td>
<td>[ 3, 4 ]</td>
</tr>
</tbody>
</table>

Ciphertext

1. $\text{Enc}_k(\text{Alice})$
2. $\text{Enc}_k(\text{Bob})$
3. $\text{Enc}_k(\text{Eve})$
4. $\text{Enc}_k(\text{Mallory})$
Communication Cost

I WANT MY INDEX

NOW!!1
Searchable Enc. Index

Client

<table>
<thead>
<tr>
<th>Client</th>
<th>Plaintext</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Alice</td>
<td>Foo</td>
</tr>
<tr>
<td>2 Bob</td>
<td>Foo</td>
</tr>
<tr>
<td>3 Eve</td>
<td>Bar</td>
</tr>
</tbody>
</table>

Server

<table>
<thead>
<tr>
<th>Server</th>
<th>Ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Enc_k(Alice, Foo)</td>
<td></td>
</tr>
<tr>
<td>2 Enc_k(Bob, Foo)</td>
<td></td>
</tr>
<tr>
<td>3 Enc_k(Eve, Bar)</td>
<td></td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H_{sk1}(0)</td>
<td>Enc_{ik1}([ 1, 2 ])</td>
</tr>
<tr>
<td>H_{sk2}(0)</td>
<td>Enc_{ik2}([ 3])</td>
</tr>
</tbody>
</table>

sk1 = H_k( search || Foo)

sk2 = H_k( search || Bar)

ik1 = H_k( index || Foo)

ik2 = H_k( index || Bar)
Searchable Enc. Index - Search

Client

Server

Index

Token | Value
--- | ---
$H_{sk1}(0)$ | $Enc_{ik1}([1, 2])$
$H_{sk2}(0)$ | $Enc_{ik2}(3)$

Ciphertext

<table>
<thead>
<tr>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Enc_k(Alice, Foo)$</td>
</tr>
<tr>
<td>$Enc_k(Bob, Foo)$</td>
</tr>
<tr>
<td>$Enc_k(Eve, Bar)$</td>
</tr>
</tbody>
</table>

sk1 = $H_k(\text{search} || \text{Foo})$

ik1 = $H_k(\text{index} || \text{Foo})$

[1, 2]
Size matters

IF YOU DON'T HIDE THE SIZE
YOU GONNA HAVE A BAD TIME
Index based - Cash et al. - Setup

Plaintext

<table>
<thead>
<tr>
<th></th>
<th>Alice</th>
<th>Foo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Bob</td>
<td>Foo</td>
</tr>
<tr>
<td>3</td>
<td>Eve</td>
<td>Bar</td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H_{sk_1}(0)$</td>
<td>$Enc_{ik_1}([1])$</td>
</tr>
</tbody>
</table>

$sk_1 = H_k(\text{search} \ || \ Foo)$

$ik_1 = H_k(\text{index} \ || \ Foo)$

occurences["Foo"] = 0
Index based - Cash et al. - Setup (contd.)

Plaintext

<table>
<thead>
<tr>
<th></th>
<th>Alice</th>
<th>Foo</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Bob</td>
<td>Foo</td>
</tr>
<tr>
<td>3</td>
<td>Eve</td>
<td>Bar</td>
</tr>
</tbody>
</table>

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H_{sk1}(0)$</td>
<td>$Enc_{ik1}([1])$</td>
</tr>
<tr>
<td>$H_{sk1}(1)$</td>
<td>$Enc_{ik1}([2])$</td>
</tr>
</tbody>
</table>

$sk1 = H_k(\text{search} \ || \ Foo)$

$ik1 = H_k(\text{index} \ || \ Foo)$

occurences["Foo"] = 1
sk2 = $H_k(\text{search} || \text{Bar})$

ik2 = $H_k(\text{index} || \text{Bar})$

occurrences["Bar"] = 0
Index based - Cash et al.

\[ \text{sk1} = H_k(\text{search} || \text{Foo}) \]
\[ \text{sk2} = H_k(\text{search} || \text{Bar}) \]
\[ \text{ik1} = H_k(\text{index} || \text{Foo}) \]
\[ \text{ik2} = H_k(\text{index} || \text{Bar}) \]
Index based - Cash et al. - Search

\[ sk_1 = H_k( \text{search} || \text{good}) \]
\[ ik_1 = H_k( \text{index} || \text{good}) \]

Client

Server

Index

<table>
<thead>
<tr>
<th>Token</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( H_{sk1}(0) )</td>
<td>( \text{Enc}_{ik1}([1]) )</td>
</tr>
<tr>
<td>( H_{sk1}(1) )</td>
<td>( \text{Enc}_{ik1}([2]) )</td>
</tr>
<tr>
<td>( H_{sk2}(0) )</td>
<td>( \text{Enc}_{ik2}([3]) )</td>
</tr>
</tbody>
</table>

Ciphertext

\[ \text{Enc}_k(\text{Alice}) \]
\[ \text{Enc}_k(\text{Bob}) \]
\[ \text{Enc}_k(\text{Eve}) \]

[1, 2]
Speed

Plaintext size (King James Bible): 4.3 MB
Ciphertext size: 4.3 MB
Time to encrypt: 0.108 sec
Time to search: 0.001 sec
Outlook
Outlook

- So far: deterministic search token → statistical analysis
So far: deterministic search token $\rightarrow$ statistical analysis

Making existing approaches practical is a challenge (e.g. FHE)
Outlook

- So far: deterministic search token $\rightarrow$ statistical analysis
- Making existing approaches practical is a challenge (e.g. FHE)
- Implement and adapt!!1
Conclusions
Conclusions

- Presented some schemes and their properties
Conclusions

- Presented some schemes and their properties
  - Det
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
  - Keyword (Song, Wagner, Perrig)
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in $O(n)$
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - Search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in $O(n)$
  - Index (Cash et al.)
Presented some schemes and their properties

- **Det**
  - Fast setup
  - Search insecure

- **Keyword (Song, Wagner, Perrig)**
  - Search is in $O(n)$

- **Index (Cash et al.)**
  - Search is in $O(1)$
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - Search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in O(n)
  - Index (Cash et al.)
    - Search is in O(1)
    - Index maintenance needed (think: Update)
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in $O(n)$
  - Index (Cash et al.)
    - Search is in $O(1)$
    - Index maintenance needed (think: Update)
- slightly different features
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in $O(n)$
  - Index (Cash et al.)
    - Search is in $O(1)$
    - Index maintenance needed (think: Update)
- slightly different features
- more exist!
Conclusions

- Presented some schemes and their properties
  - Det
    - Fast setup
    - search insecure
  - Keyword (Song, Wagner, Perrig)
    - Search is in O(n)
  - Index (Cash et al.)
    - Search is in O(1)
    - Index maintenance needed (think: Update)
- slightly different features
- more exist!
- Searching on encrypted data is practical
References:
