A Secure Searcher for End-to-End Encrypted Email

Balamaruthu Mani

Supervisor: Professor Barak A. Pearlmutter
Introduction

Email Communication
Encryption over network

Sender

Mail Server

Plain email text

Recipient

Encrypted on transfer
End-to-End Encryption

Email Communication
End-to-End Encryption

Client1

Mail Server

Encrypted on transfer

Encrypted email text

Mail Server

Client side Key and Index management

Client2
Motivation

- Client side End-to-End Encrypted email is needed to ensure privacy

- Current tools or extensions that support encrypted email do not have a direct support for search

- A naive solution would be to decrypt all messages and store them on the local machine
  - Security risk if the local machine is compromised
  - Inefficient as it is not accessible from other machines
Solution

User 1

Local Machine 1
- Email Client
- Secure Searcher
- Email Index

Cloud Untrusted Storage
- Email Index

User 2

Local Machine 2
- Email Client
- Secure Searcher
- Email Index

End-to-End Email Encryption
Secure Index – Technique

- Build a secure index using Goh’s Bloom filter technique where Bloom filter is a data structure with bit vector as a base.

- Catch: There is a false positive rate \( fp = (1/2)^r \) where \( r = (\ln 2)(m/n) \) based on the number of elements to be mapped \( n \), size of the Bloom filter \( m \) and the number of hash functions used \( r \).
Secure Index – Indexing

-------BEGIN PGP MESSAGE-------
Version: Mailvelope v0.13.1
Comment: https://www.mailvelope.com

wcFMA0VxwhGntIuAQ/ +Kk7+euGZcpg1au2eOgBODBZQdMI0maLaba3g/zM9JbwOcpX2z3nMyzz6Ba73aoBczASvXkGEwVsQVioOawgns4NSXnbQgsyJhNer7c2ERr/MDSPjIwdVf+TfhTp5ddxqf/uOGiKVk4nXM6n0PZ1WVQBtn7GqSTknIKbNjZ3bMoSw4qLCPezISWXXYBwby35ZQIKz3fY0pdhsAURjPHXuYZIAWvBIqEH5uTCeCSBhyz0yY8o+b52UGpcdLs7BcHrXFhT8xZGhgRE1V7DflyW+55nDIPf2tF/89KiMTBHUMKN7uE3S2T37TC3F4rmn1bYAX2tCD4Ew1IZCgx9b4ftanvg+YuwrqTZStvgm9CDCx3wWNigqjm2GgWR+UVsoSdq0C6pDM4YjFINVRrynIWXmmw40fzDWeCyGg+sEWmGaUwylcxEtJUhmeOMuFomZkUsDLkfRfCpnnNQGLHsk7A9iCeY7bb2H66O8kEP38VSz95aUMJfplUv5CED4csKyXz0sHSPOAplc7OYrLsxyKVRiNRkyKr7MxvpmB5B4Y+qYH374v46ncAnbtN+QlPfmU3mAw7i2HnjDoWY9tRV4WRXXemR6OUTDH0iYcYeSY23TZA2S3wiOc6m6GGBZMjxp7dYzIkF2Wkn9RED8I/LyNDX4vFoEW7c5q81yVQLDSUgFyqo0iAtCu2pxA
CF/uQGZbGXR+GiGr18IH+xtGX/uhHq/hYN9kZWE1rR6Ypo1CnAbFBUZMVInn7vZMGa1lzG6CzLwEzecPSfOAedJzyfCbHWl=
=Qj0J
-------END PGP MESSAGE-------
Secure Index – Indexing

- 1 (Message Id)
- This is test pgp (Decrypted Message Body)

- For each word in Message
- This
- Hash Function
- Indexing Key
- Tw as key
- Message Id (1)
- Hash Function
- Message Id
- Bloom Filter
- Set mapped indices
- Secure Index
- store
Secure Index – Searching

Search Word (pgp) → Hash Function → Trap door (Tw) → Hash Message Id → Check indices in bloom filter → Message Id

Retrieve

Secure Index

Bloom Filter

Hash Function

Message Id

Tw as Key
Implementation

- Implemented a library in Java using Bouncy Castle cryptographic libraries

- Exposed APIs
  - Index plain/encrypted messages
  - Search for word/words in the indexed messages
  - Import Open PGP and S/MIME private keys to Key store

- Integrated the library with Columba email client to demonstrate the secure search library usage
Columba Client Integration

Email Client
- Get All Messages
- New Message
- Message Listener
- Messages deleted

Secure Searcher
- Email Indexer Client
- Add Index
- Remove Index
- Email Index

1. Get All Messages
2. New Message

Messages deleted
Columba Client Integration

Email Client
- Get Search Word
- Display Search Results
- Email Store

Secure Searcher
- Email Searcher Client
- Email Index

1. Get Search Word
2. Display Search Results
3. Email Store
Security

- Technique is IND–CKA semantically secure (Indistinguishability under Chosen Keyword Attack)
- Security of the implementation depends on the hash function used (Default: Hmac sha256)
- Ensures confidentiality but not integrity and authenticity
- Not secure against in-memory attacks on client that decrypts the message
- Encrypting the message id may provide more security at the expense of search speed
Evaluation

- Space efficiency vs False positive (Default: 2% with 6 hash functions and 8 bits per word).
  - False positives can be removed by storing all the encrypted words

- Performance – Time
  - DaCapo Benchmark suite – luindex a corpus of 1230 text documents converted to Open PGP and S/MIME encrypted email messages
  - Indexing time is approximately 1 second for indexing 3000 words up to a maximum of 5.5 seconds for 32000 words
  - Search time in terms of number of email messages indexed in the database (3.5 seconds for searching 3690 indexes)
Conclusion

- Columba email client with integrated secure searcher is available as a runnable jar
- Columba client and secure searcher library are extensible with plug-in architecture and layered architecture respectively
- Facebook enables people to add Open PGP public keys to their profile to enhance the privacy of email messages from Facebook!
Thank you