Hiding the access pattern is not enough:

Exploiting search pattern leakage in Searchable Encryption

Simon Oya
Florian Kerschbaum

University of Waterloo
CrySP
Searchable encryption allows a user to send her encrypted database to a server while still being able to perform secure searches over it.

Example:

```
Search Index

DOG : 1, 3
CAT : 1
Cow : 2
```

Service provider = (Passive) Adversary
Efficient searchable encryption schemes leak the **search pattern** (which can be used to compute the frequency of each query token) and the **access pattern** (which reveals how many documents match the query).
We propose a new attack (SAP) that uses both search and access pattern leakage, as well as auxiliary information (which is not necessarily ground-truth information).
To derive the attack, we build a mathematical model of the response volume and frequency of each query, based on the auxiliary information.
The attack finds the maximum likelihood matching of keywords to query tokens given the previous mathematical model. It uses the Hungarian Algorithm to find the optimal matching.
SAP is easy to adapt against different volume-hiding defenses (padding) by just taking the defense into account in the mathematical model.

CLRZ: Chen et al. “Differentially private access patterns for searchable symmetric encryption”. INFOCOM’18
PPYY: Patel et al. “Mitigating leakage in secure cloud-hosted data structures”. CCS’19
We evaluate SAP using real datasets (Enron and Lucene) and use query frequencies grabbed from Google Trends. We give the adversary imperfect auxiliary information to run the attack.
Results I

5 queries/week
50 weeks

Importance of search+access pattern

By combining both volume and frequency information, SAP achieves high query recovery (left). SAP outperforms current state-of-the-art attacks (right).


graphm: Pouliot and Wright. “The shadow nemesis: Inference attacks on efficiently deployable, efficiently searchable encryption”. CCS’16
By adapting SAP against previous defenses, we are able to practically bypass two of them (CLRZ and PPYY) and we still achieve non-trivial recovery rates for the third one (SEAL).
Conclusions

- SAP is efficient and strong
- Frequency + Volume leakage is dangerous
- Padding strategies are not very effective
- Hiding search pattern and/or frequency is important

Our results show the importance of hiding search patterns and/or frequency leakage. Recent works that are moving in this direction seem promising.

There is hope!

- PANCAKE: Grubbs et al. USENIX'20
- OSSE: Shang et al. NDSS'21
- SW:SSSE: Gui et al.

Thank you!

Simon.oya@uwaterloo.ca