CWI
Database Architectures

Vectorized Query Processing over encrypted data

MSc Research Project

Sam Ansmink (sam@cwi.nl)

Supervisor: Peter Boncz (VU, CWI)
Second reader: Marc Makkes (VU)

VU VRIJE UNIVERSITEIT AMSTERDAM

UNIVERSITY OF AMSTERDAM

Query processing on encrypted data

Paradigm shift: cloud computing

- Secure outsourced databases
- First described in 2002[1]

New threat model

- Untrusted server
  - Curious cloud providers
  - Malicious governments
  - Compromised cloud infrastructure
- Trusted client

Query processing on encrypted data

Operate directly on encrypted data

- Homomorphic encryption
- Property preserving encryption
- Searchable encryption
- Secure multiparty computation

Create a trusted "zone" on the untrusted server

- Secure Coprocessor (SCPU), FPGA
- Intel SGX, ARM Trustzone, AMD SEV, Microsoft VBS

Query processing on encrypted data

Operate directly on encrypted data

- Homomorphic encryption
- Property preserving encryption
- Searchable encryption
- Secure multiparty computation

Create a trusted "zone" on the untrusted server

- Secure Coprocessor (SCPU), FPGA
- Intel SGX, ARM Trustzone, AMD SEV, Microsoft VBS

Existing literature on EDBMS

Trusted Execution Environment (TEE)

- OLTP: StealthDB[2], EnclaveDB[3], SQL Server AEv2[4]
- OLAP: Opaque[5], ObliDB[6], EncDBDB[7]

Our contribution

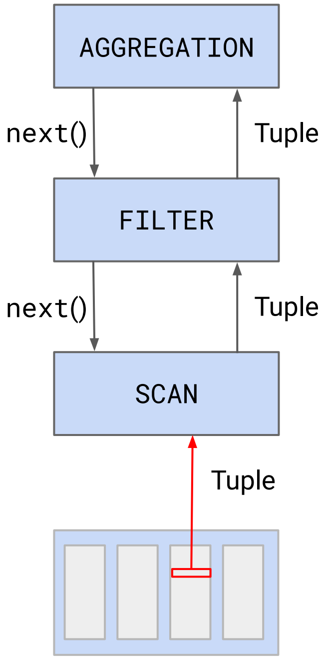
- Use of vectorized query engine
- Focus on high efficiency

Research goal

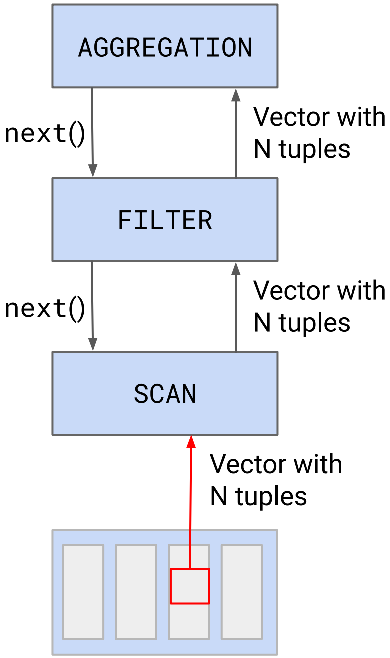
Design EDBMS prototype

- DuckDB and Intel SGX
- Vectorized query execution
- Focus on minimizing overhead

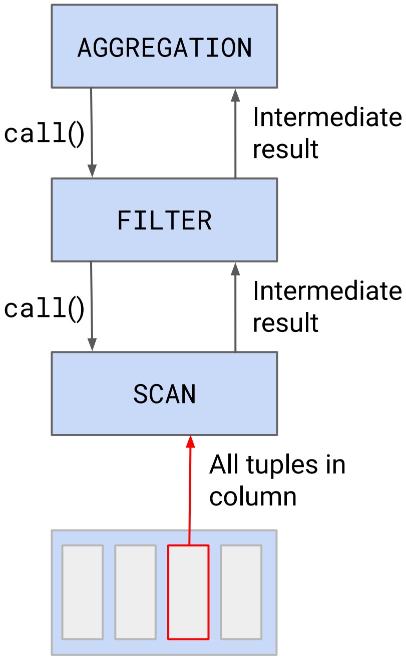
Query execution models



Tuple-at-a-time



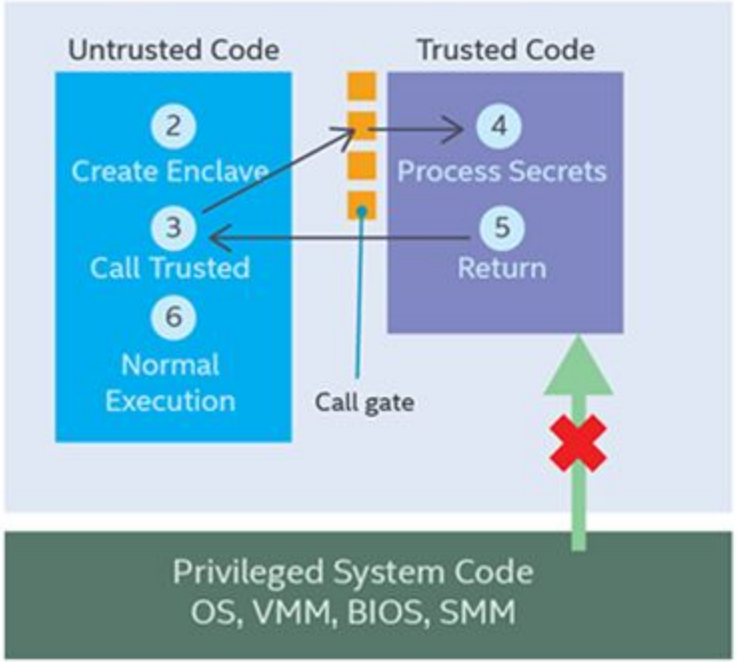
Vectorized



Column-at-a-time

Intel SGX

- Hardware enforced “enclaves”
- Split codebase (secure/unsecure)
- Split data (secure/unsecure)



Performance cost of Intel SGX

Limited secure memory

- ~172MB on 10th gen Intel
- ~96MB on 6th - 9th gen Intel

Performance critical factors

- Secure memory paging
- Enclave-mode entry/exit (~ 1000 - 16000 cycles)
- Access to secure memory (CPU cache misses)

Overhead of decryption

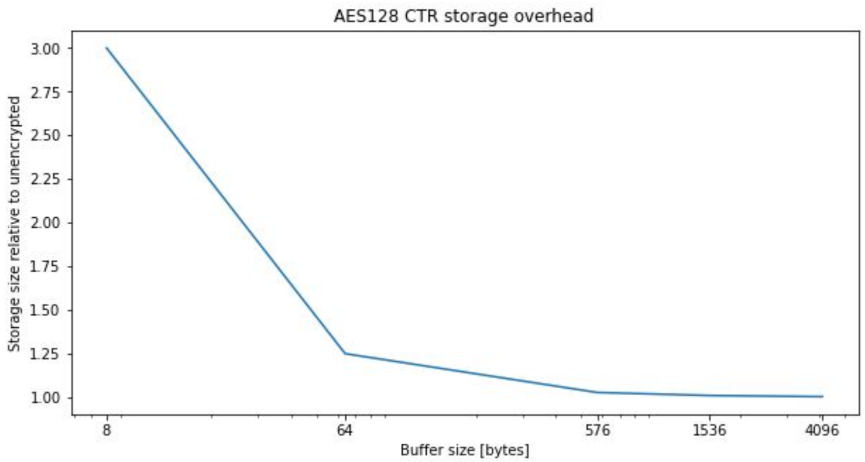
Storage cost

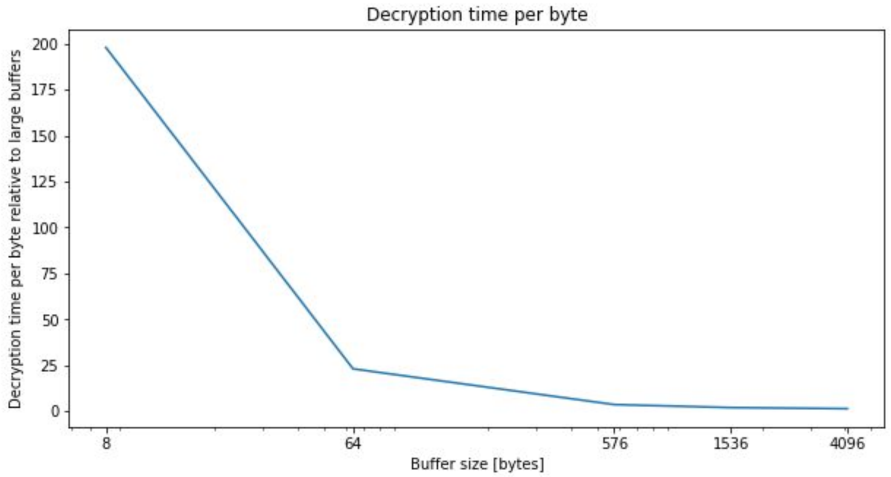
- Extra data to store (e.g. initialization vector)
- Encrypted data has poor compression

Computational cost

- Depends strongly on buffer size

Overhead of decryption





SGX-based EDBMS design

Vectorized execution matches requirements well

- No large materialization
- Easily amortize encryption overhead
- Prevent excessive enclave entries

SGX-based DBMS design

Which parts to run in enclave?

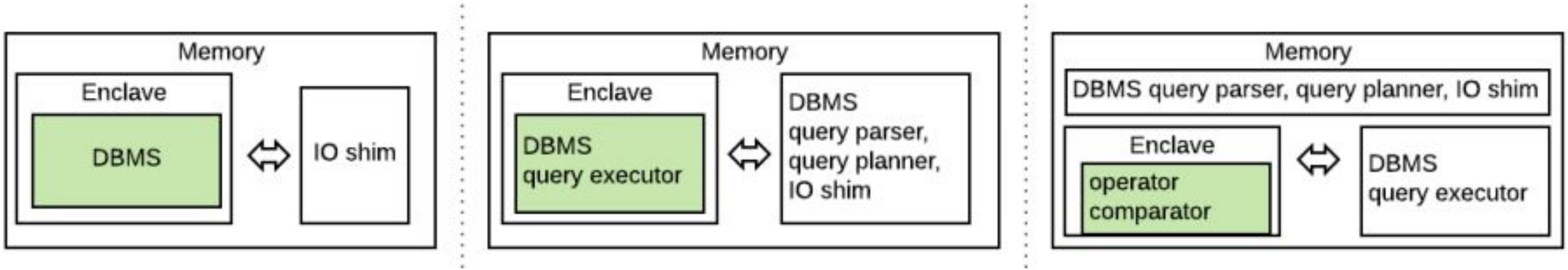
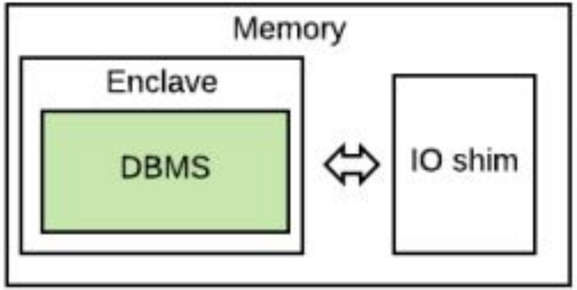


Image source: StealthDB[1]

Two designs tested

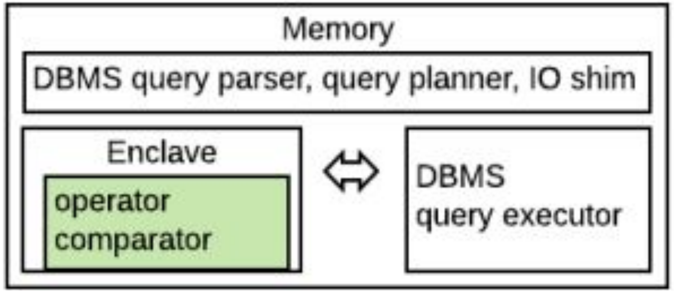
Model 1: Graphene SGX

- Using Graphene-SGX
- Whole DBMS in enclave



Model 2: SGX SDK

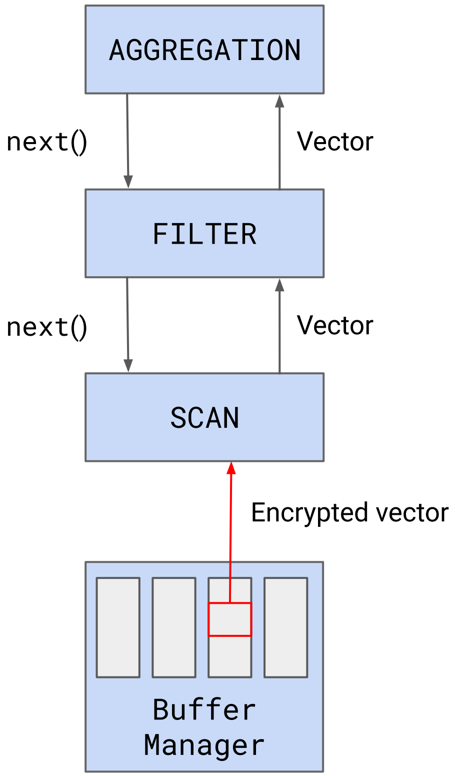
- Using SGX SDK
- Operators in enclave



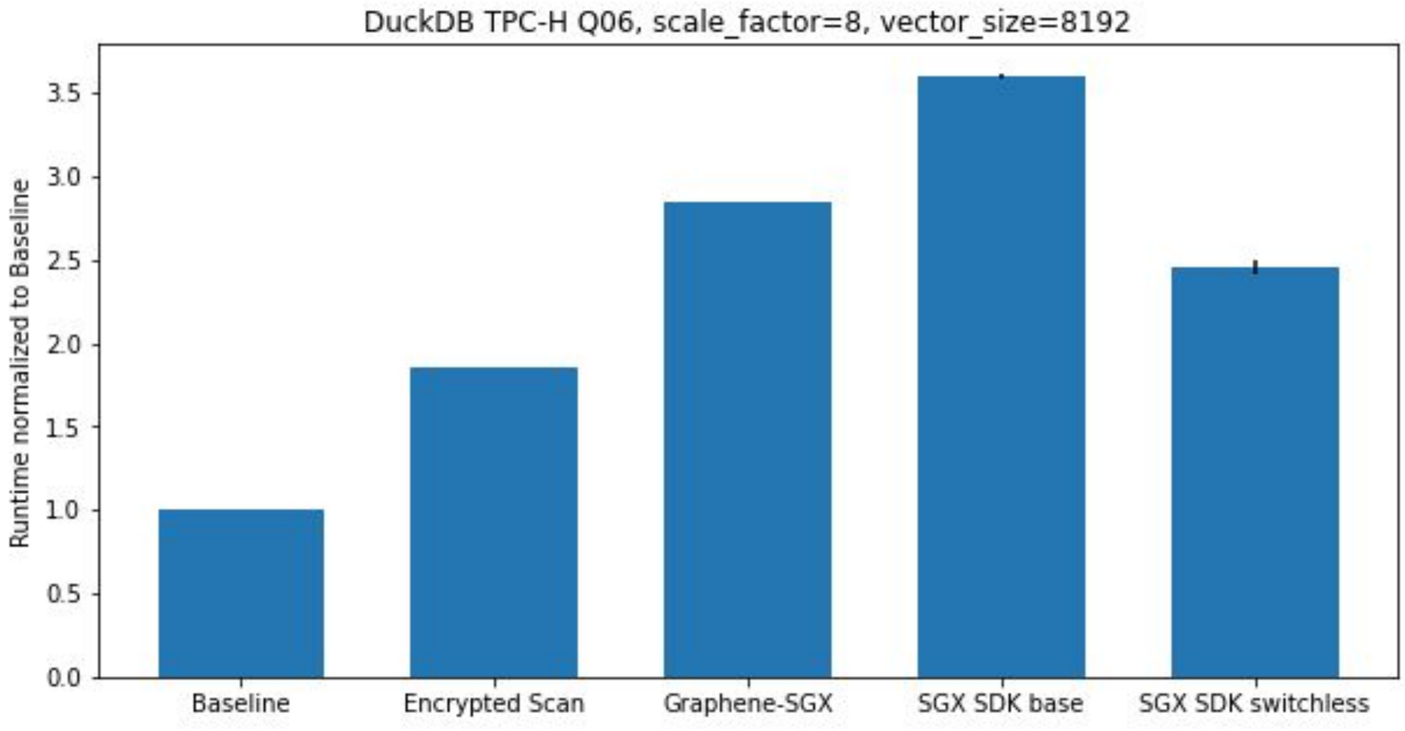
Baseline Encrypted Implementation

Encrypted Scan

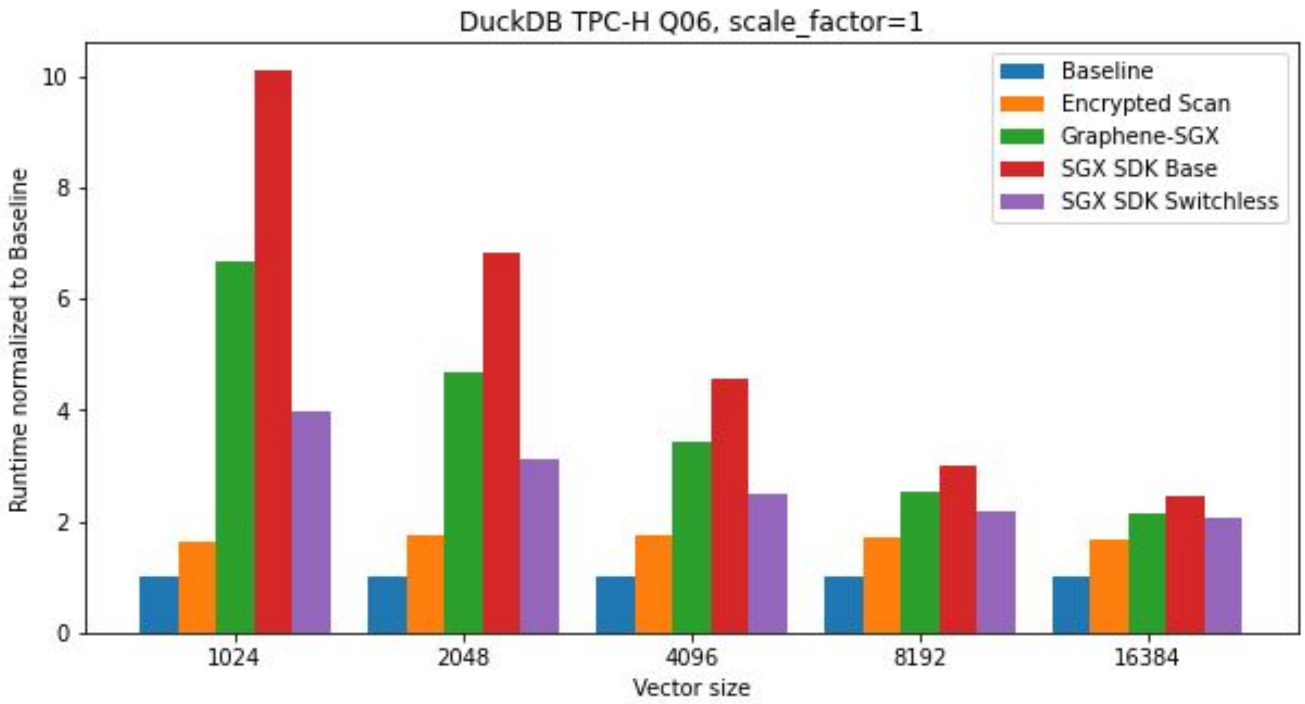
- Data encrypted per vector
- Decryption in scan operator
- Fixed length data-types only (no strings yet)



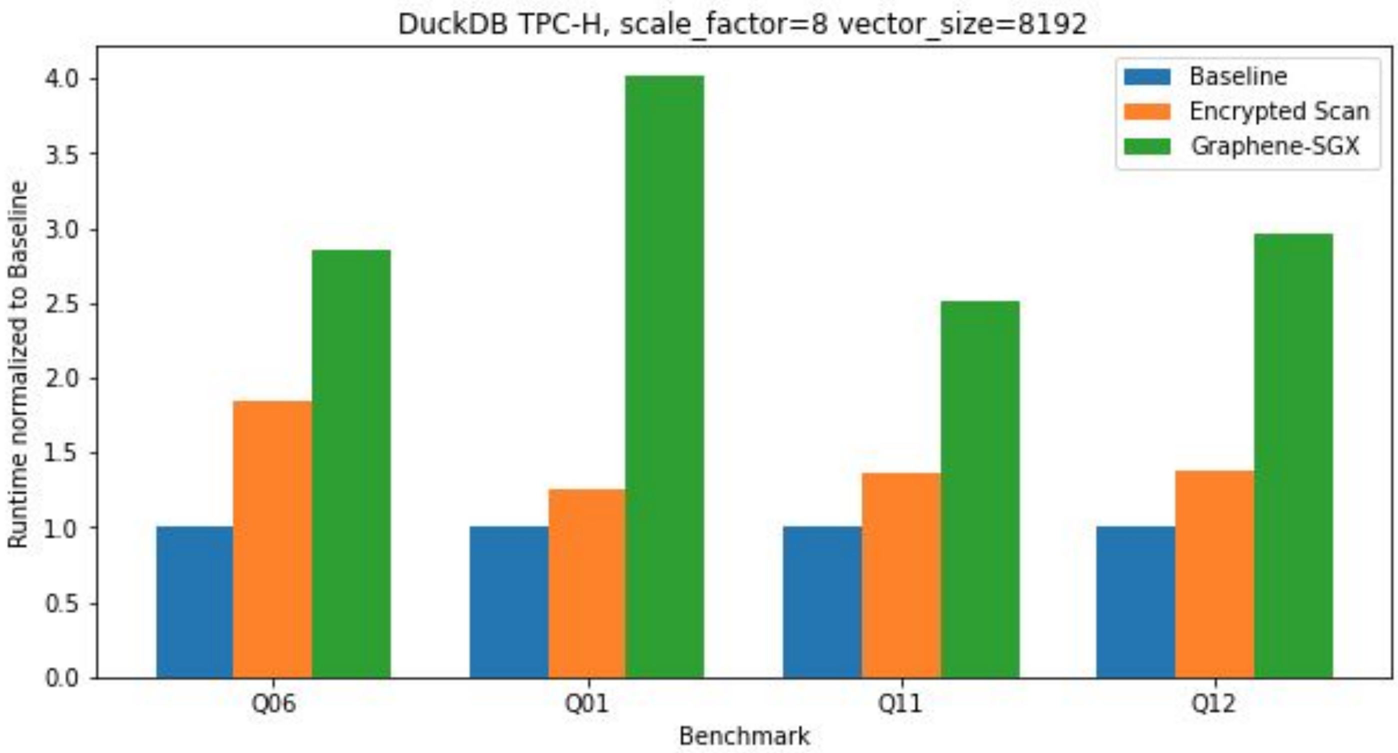
Results: Overview



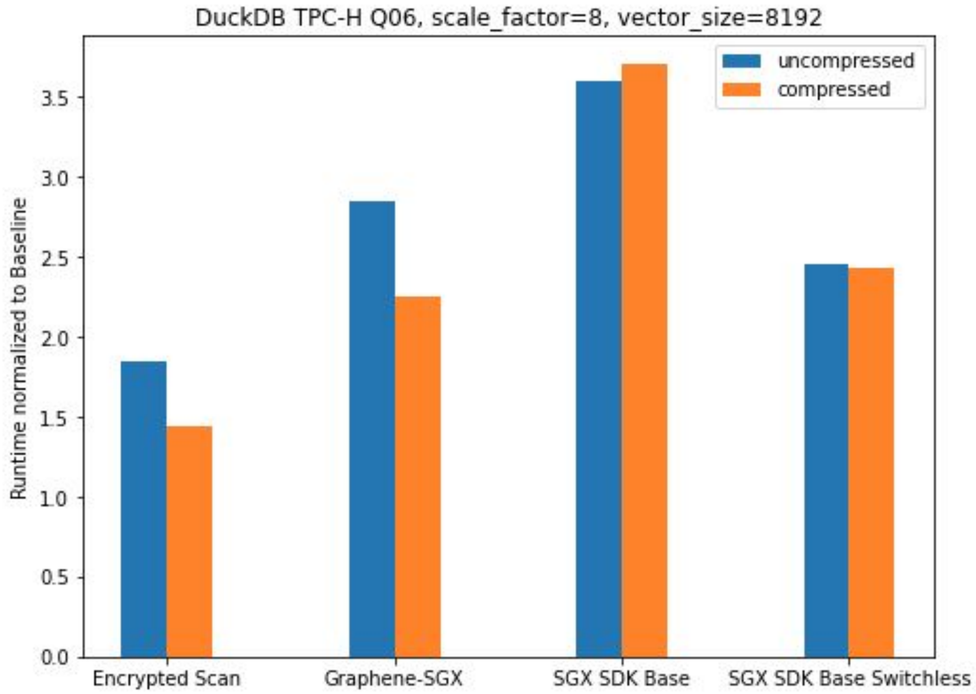
Results: Impact of vector size



Results: Graphene-SGX



Results: Effect of compression



- Compressed execution
- Compression ratio: 3x
- SGX SDK implementation suffers from extra enclave entries

Conclusions

- Vectorized execution fits SGX model well
- Low overhead encrypted query processing
- Both models analyzed are feasible

Future work

- Support (efficient) joins
- Support string data (see encDBDB[7])
- Oblivious execution (see ObliDB[6])
- Other TEEs (e.g. ARM Trustzone)

CWI Database Architectures

References

(1) Hacigümüş, Hakan, et al. "Executing SQL over encrypted data in the database-service-provider model." *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*. 2002.

(2) Gribov, Alexey, Dhinakaran Vinayagamurthy, and Sergey Gorbunov. "Stealthdb: a scalable encrypted database with full sql query support." *arXiv preprint arXiv:1711.02279* (2017).

(3) Priebe, Christian, Kapil Vaswani, and Manuel Costa. "EnclaveDB: A secure database using SGX." *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018.

(4) Antonopoulos, Panagiotis, et al. "Azure SQL Database Always Encrypted." *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 2020.

(5) Zheng, Wenting, et al. "Opaque: An oblivious and encrypted distributed analytics platform." *14th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 17)*. 2017.

(6) Eskandarian, Saba, and Matei Zaharia. "ObliDB: oblivious query processing using hardware enclaves." *arXiv preprint arXiv:1710.00458* (2017).

(7) Fuhry, Benny, and Florian Kerschbaum. "Encdbdb: Searchable encrypted, fast, compressed, in-memory database using enclaves." *arXiv preprint arXiv:2002.05097* (2020).

Questions?