Accelerating Encrypted Data Stores Using Programmable Switches

Carson Kuzniar, Miguel Neves, Israat Haque

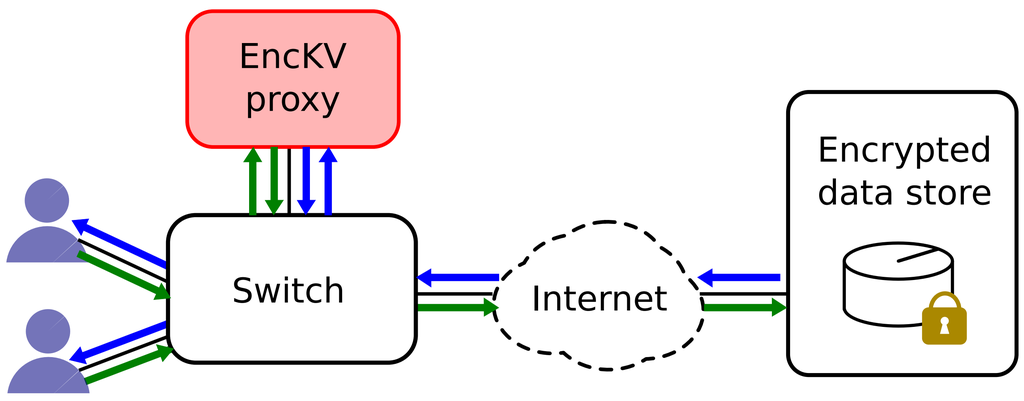
Problem Statement

- Senstive information prevents some applications from using cloud resources.
- Using encrypted data storage protects privacy, but comes with a performance cost.

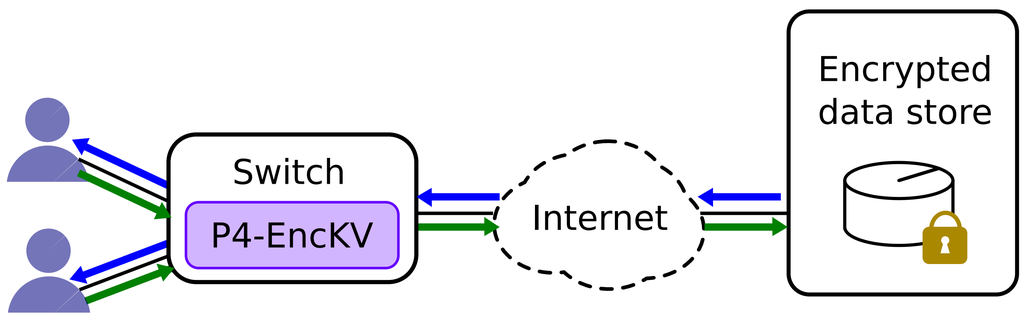
Approach

- We propose using programmable network devices to acclerate encrypted queries.

System Overview



- Encypted storage relies on additional processing at the client or a client proxy.
- Peformance is often limited by the proxies capabilities.
- Moving operations to the network helps alleviate the proxy bottleneck.
- In-network processing removes the travel time to the proxy.
- Switches are optimized for high-speed concurrent processing reducing slowdown due to congestion.



Testbed

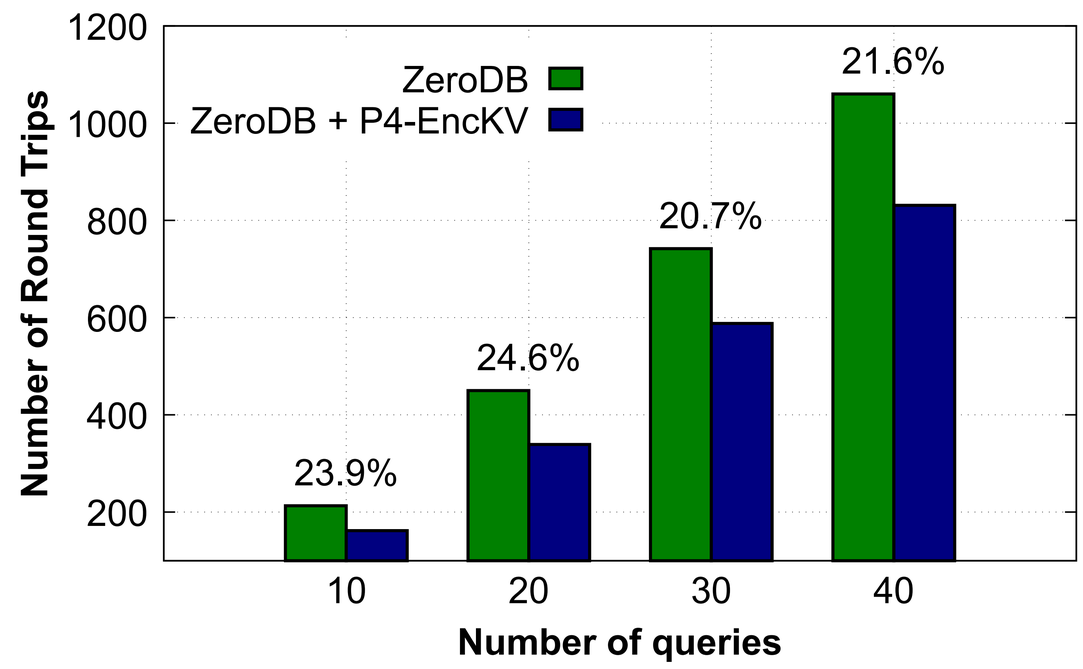
- P4EncKV prototype using BMv2 software switch.
- Client-server network emulated in Mininet.
- Hosts running ZeroDB encrypted database.
- Indexes are stored as BTrees whose nodes are cached in the network.
- Nodes are decrypted by the client.

Example Operation: Caching

- Keys are stored with a hashmap and a blockmap.
- Blockmaps indicates which blocks contain information related to a desired key.
- Hashmaps indicate which algorithm to use for retrieving a block index.
- Block arrays are contructed in powers of two to serve values of any size.

Preliminary Results

- Experiments performed on an encrypted database with 5K entries (random integers).
- Our results show a 20-25% improvement in total round trips.



Next Steps

- Test solution on hardware switches.
- Expand P4EncKV to support other operations used in encrypted storage.